(12) UK Patent Application (19) GB (11) 2 336 699 (13) A

(43) Date of A Publication 27.10.1999

(21) Application No 9808807.3

(22) Date of Filing 24.04.1998

(51) INT CL[6]
G06F 17/30

(52) UK CL (Edition Q)
G4A AUDB

(56) Documents Cited
EP 0822502 A1 EP 0364179 A2 US 5659766 A
US 5598557 A

(58) Field of Search
UK CL (Edition P) G4A AUDB
INT CL[6] G06F 17/30

(71) Applicant(s)
The Dialog Corporation Plc
(Incorporated in the United Kingdom)
The Communications Building, 48 Leicester Square,
LONDON, WC2H 7DB, United Kingdom

(72) Inventor(s)
Llewelyn Ignazio Fernandes
Rachel Hammond

(74) Agent and/or Address for Service
Atkinson & Co
First Floor, Unit A, The Technology Park,
60 Shirland Lane, SHEFFIELD, S9 3PA,
United Kingdom

(54) Abstract Title
Automatic classification of text files

(57) Text files are automatically categorised by examining the files for occurrences of keywords indicative of a category. A score is adjusted in response to the occurrences of the keywords, and a classification of the text file to a category is made if the score exceeds a threshold. Because longer files will tend to produce more occurrences than short files of equal relevance, the scores for files shorter than a certain length T are increased by a weighting factor K which is inversely related to the file size S. Plural words and phrases may be sought in each file and the scores accumulated in dependence on the occurrence of these terms. A branching hierarchical file assessment structure may be used, in which case score adjustment may be performed by changing scoring coefficients stored at branches (figures 6-8).

"level of expertise", even if it was a "category", is not determined using keywords in our invention

FILE WEIGHTING FACTOR, W, IS GIVEN BY :

IF $S < T$

$$W = \left(\frac{T - S}{T}\right) N + 1$$

ELSE

$$W = 1$$

WHERE T = THRESHOLD VALUE
S = SIZE OF CURRENT FILE
N = WEIGHTING CONSTANT, IE. 0.8

IE.

WHEN $S < 2048$, $W = \left(\frac{2048 - S}{2048}\right) 0.8 + 1$

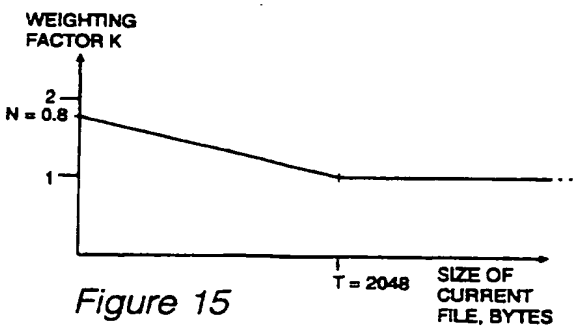

Figure 15

GB 2 336 699 A

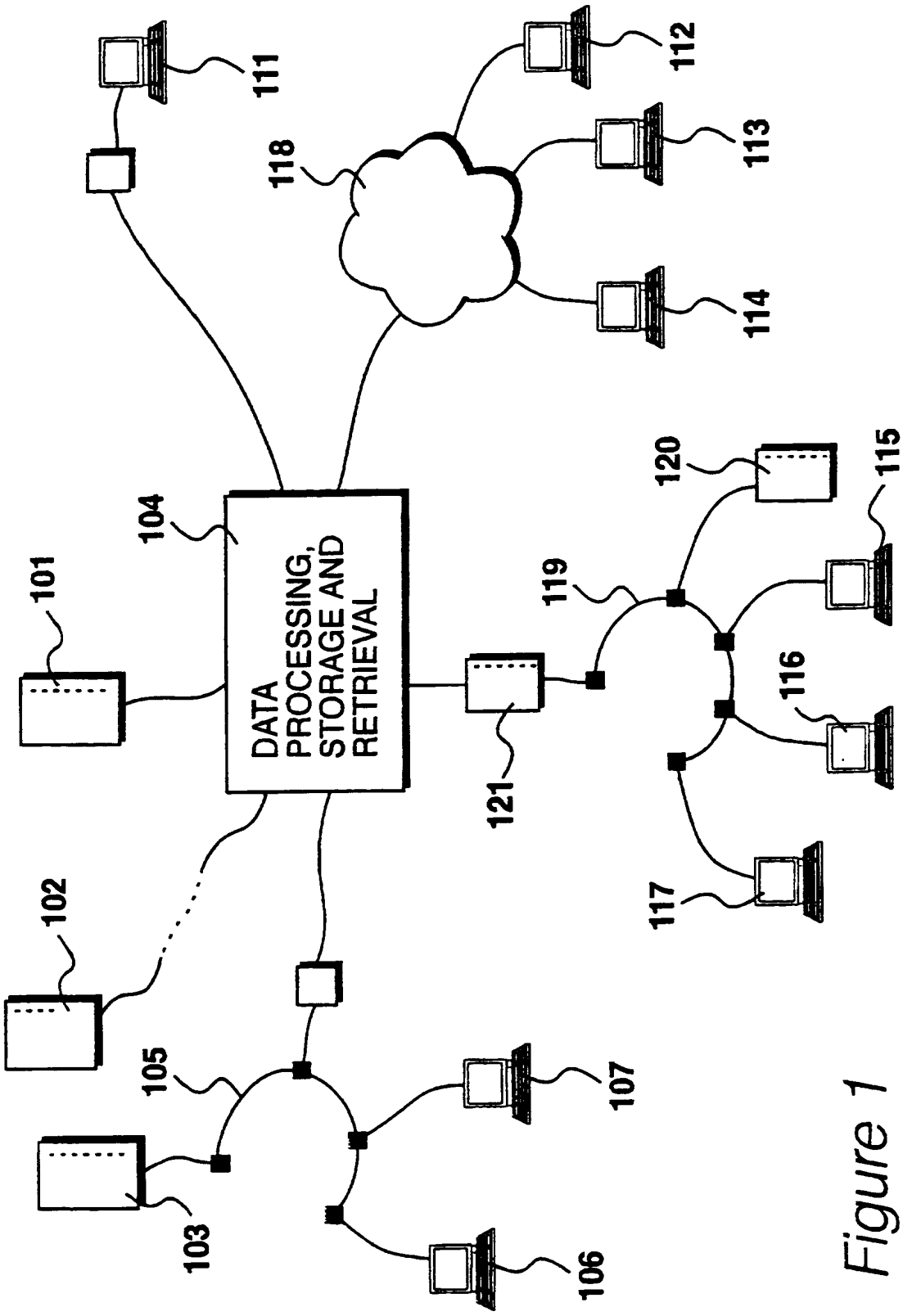
DATA PROCESSING, STORAGE AND RETRIEVAL
101
102
103
104
105
106
107
111
112
113
114
115
116
117
118
119
120
121

Figure 1

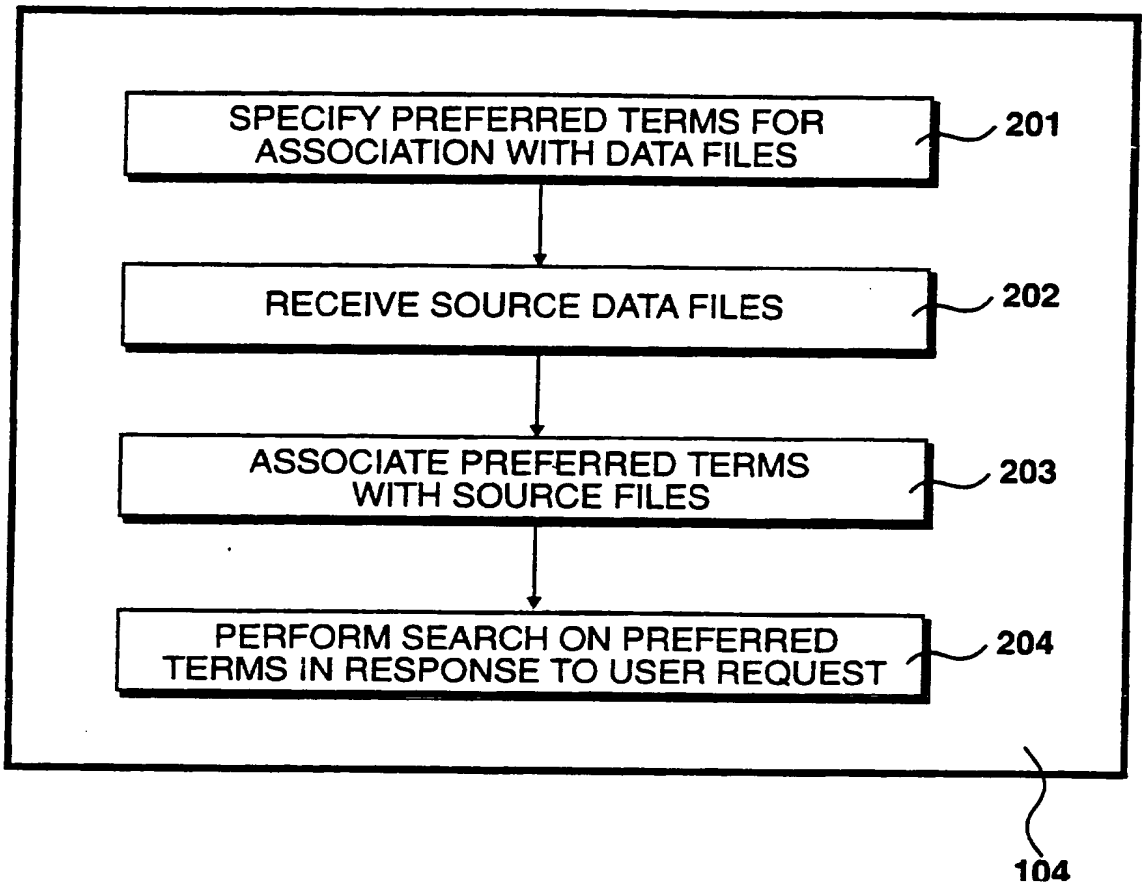
SPECIFY PREFERRED TERMS FOR ASSOCIATION WITH DATA FILES
201
RECEIVE SOURCE DATA FILES
202
ASSOCIATE PREFERRED TERMS WITH SOURCE FILES
203
PERFORM SEARCH ON PREFERRED TERMS IN RESPONSE TO USER REQUEST
204
104

Figure 2

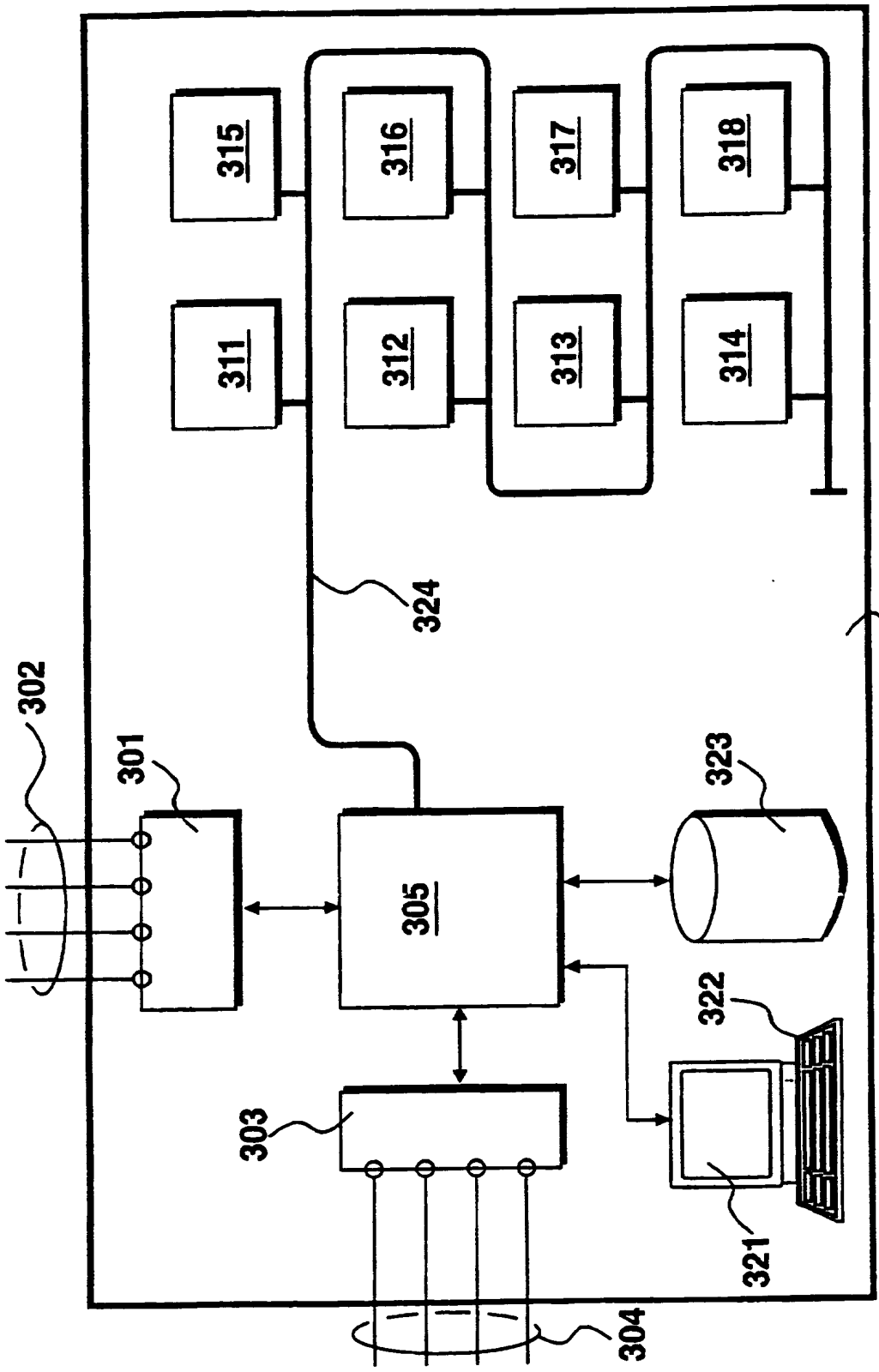
315
316
317
318
311
312
313
314
324
104
302
301
323
305
322
303
321
304

Figure 3

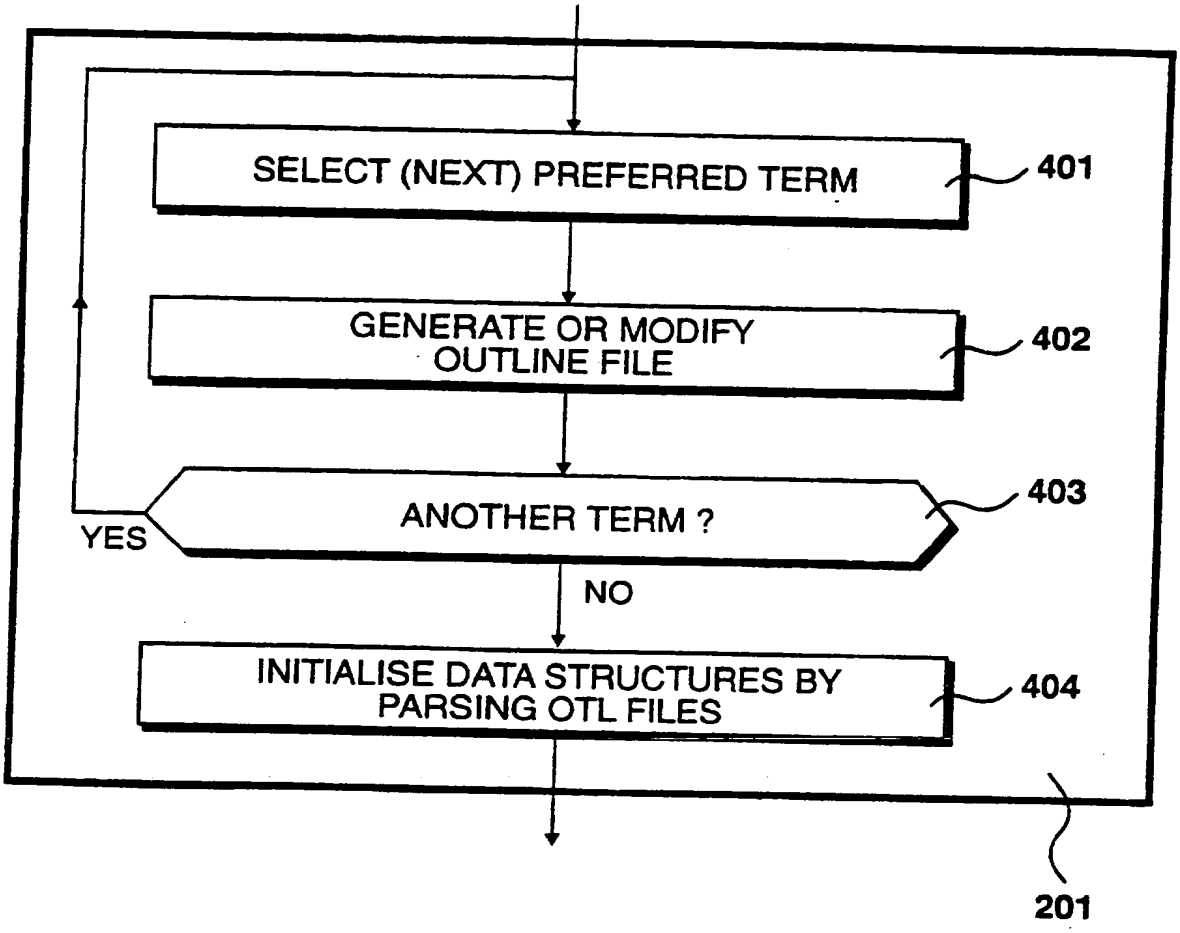
SELECT (NEXT) PREFERRED TERM
401
GENERATE OR MODIFY OUTLINE FILE
402
ANOTHER TERM ?
403
YES
NO
INITIALISE DATA STRUCTURES BY PARSING OTL FILES
404
201

Figure 4

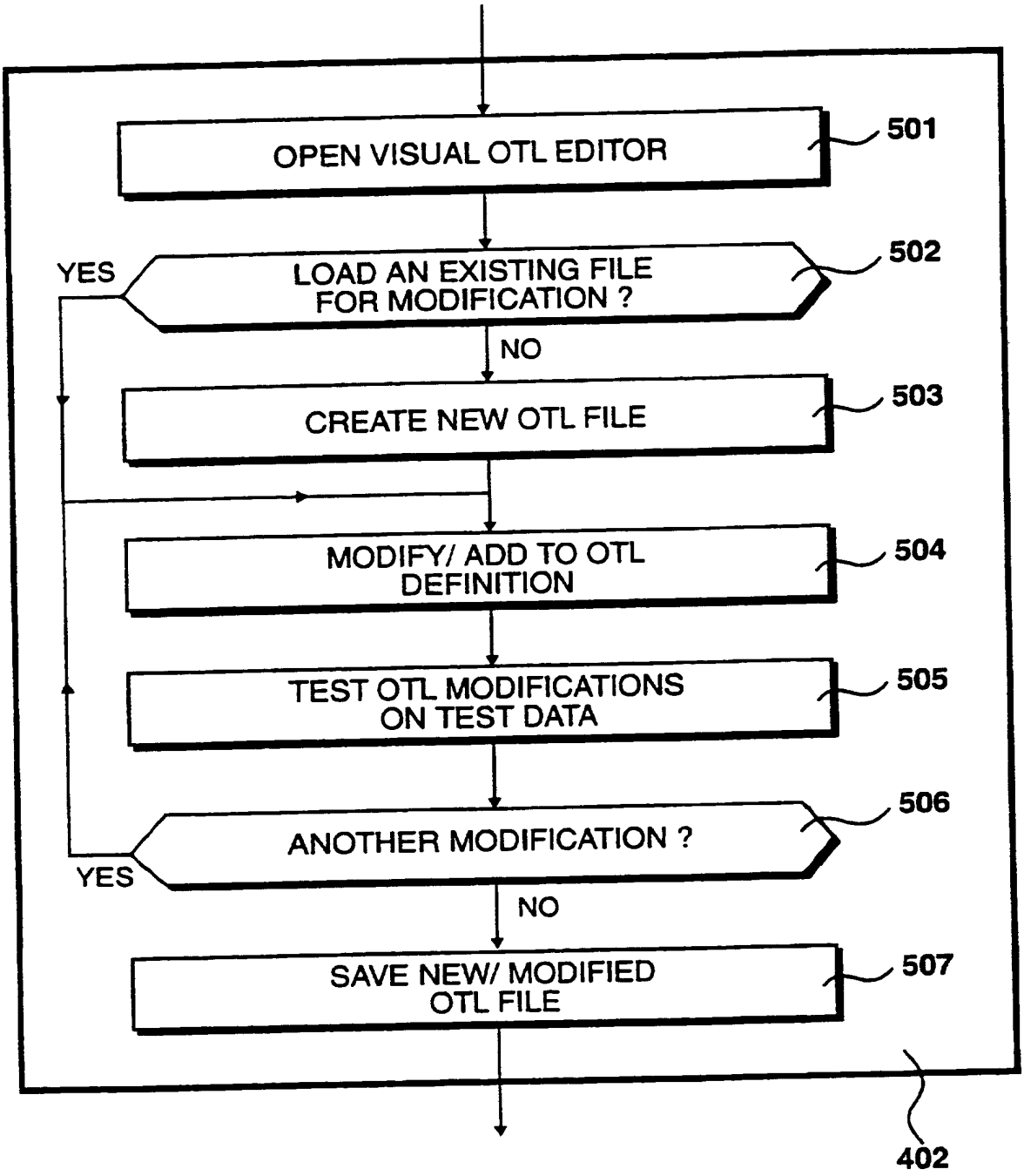
OPEN VISUAL OTL EDITOR
501
YES
LOAD AN EXISTING FILE FOR MODIFICATION ?
502
NO
CREATE NEW OTL FILE
503
MODIFY/ ADD TO OTL DEFINITION
504
TEST OTL MODIFICATIONS ON TEST DATA
505
ANOTHER MODIFICATION ?
506
YES
NO
SAVE NEW/ MODIFIED OTL FILE
507
402

Figure 5

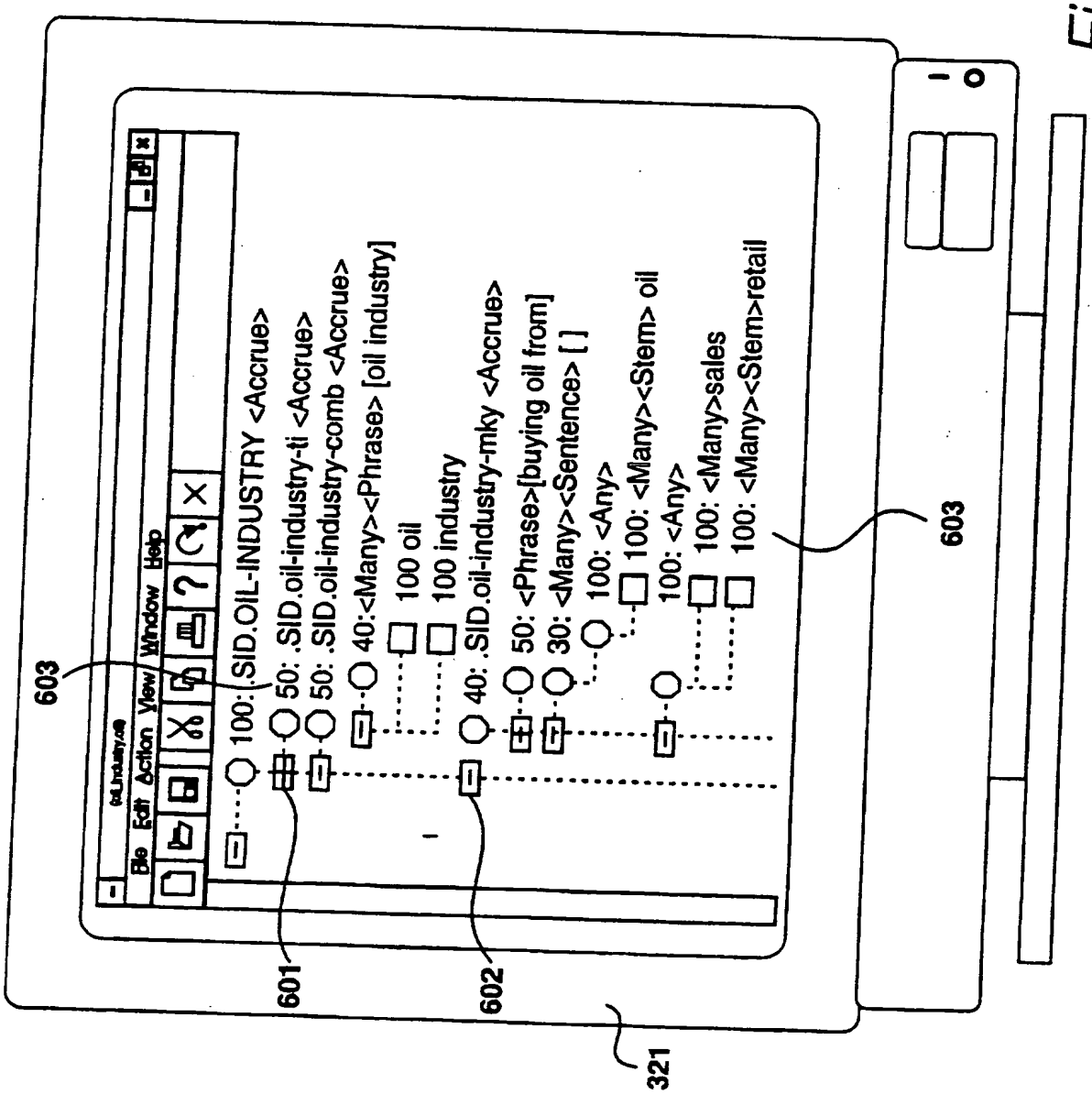
603
601
602
603
321
File Edit Action View Window Help
100: .SID.OIL-INDUSTRY <Accrue>
50: .SID.oil-industry-ti <Accrue>
50: .SID.oil-industry-comb <Accrue>
40:<Many><Phrase> [oil industry]
100 oil
100 industry
40: .SID.oil-industry-mky <Accrue>
50: <Phrase>[buying oil from]
30: <Many><Sentence> [ ]
100: <Any>
100: <Many><Stem> oil
100: <Any>
100: <Many>sales
100: <Many><Stem>retail

Figure 6

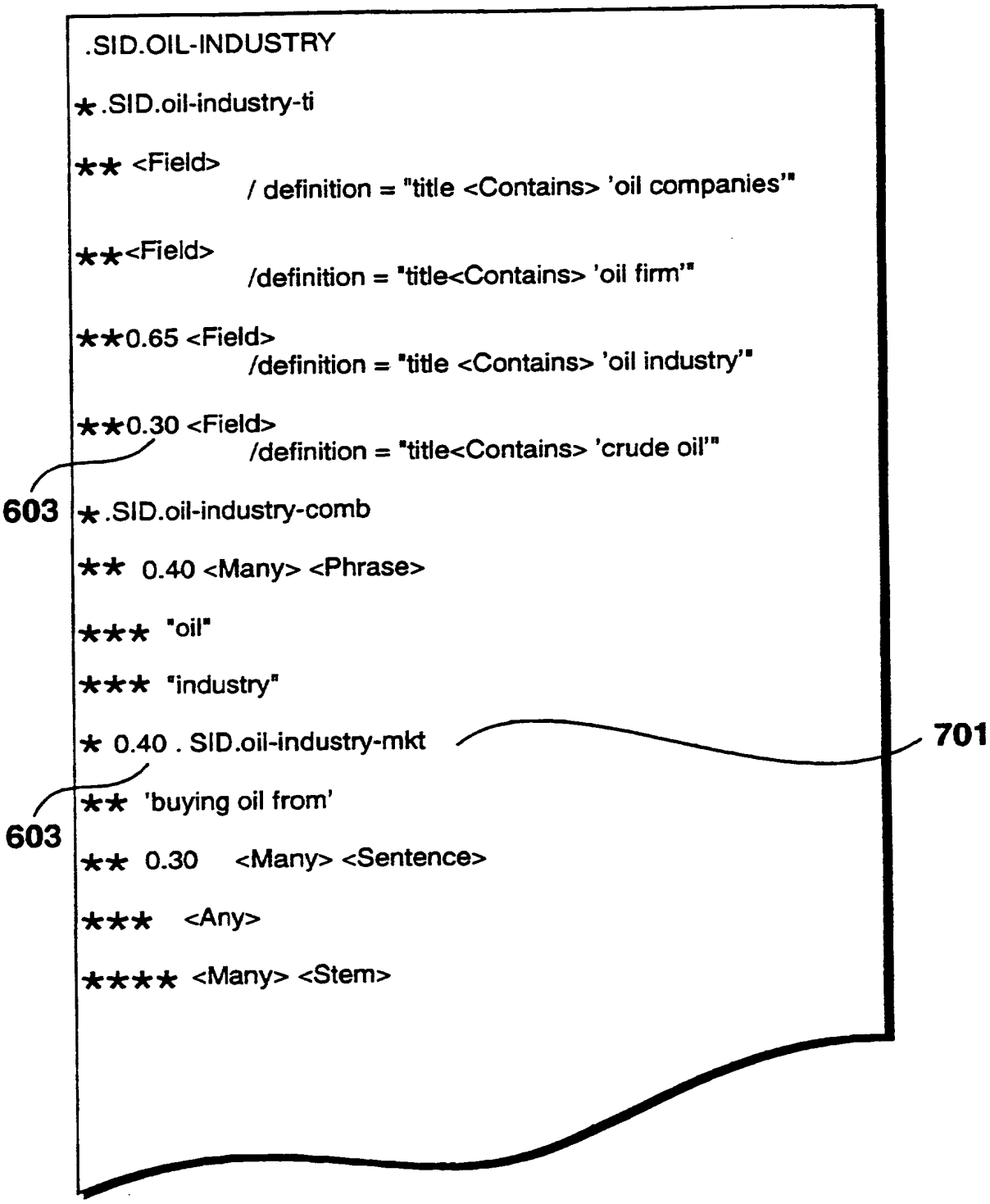
.SID.OIL-INDUSTRY
*.SID.oil-industry-ti
** <Field>
/ definition = "title <Contains> 'oil companies'"
**<Field>
/definition = "title<Contains> 'oil firm'"
**0.65 <Field>
/definition = "title <Contains> 'oil industry'"
**0.30 <Field>
/definition = "title<Contains> 'crude oil'"
603
*.SID.oil-industry-comb
** 0.40 <Many> <Phrase>
*** "oil"
*** "industry"
* 0.40 . SID.oil-industry-mkt
701
** 'buying oil from'
603
** 0.30 <Many> <Sentence>
*** <Any>
**** <Many> <Stem>

Figure 7

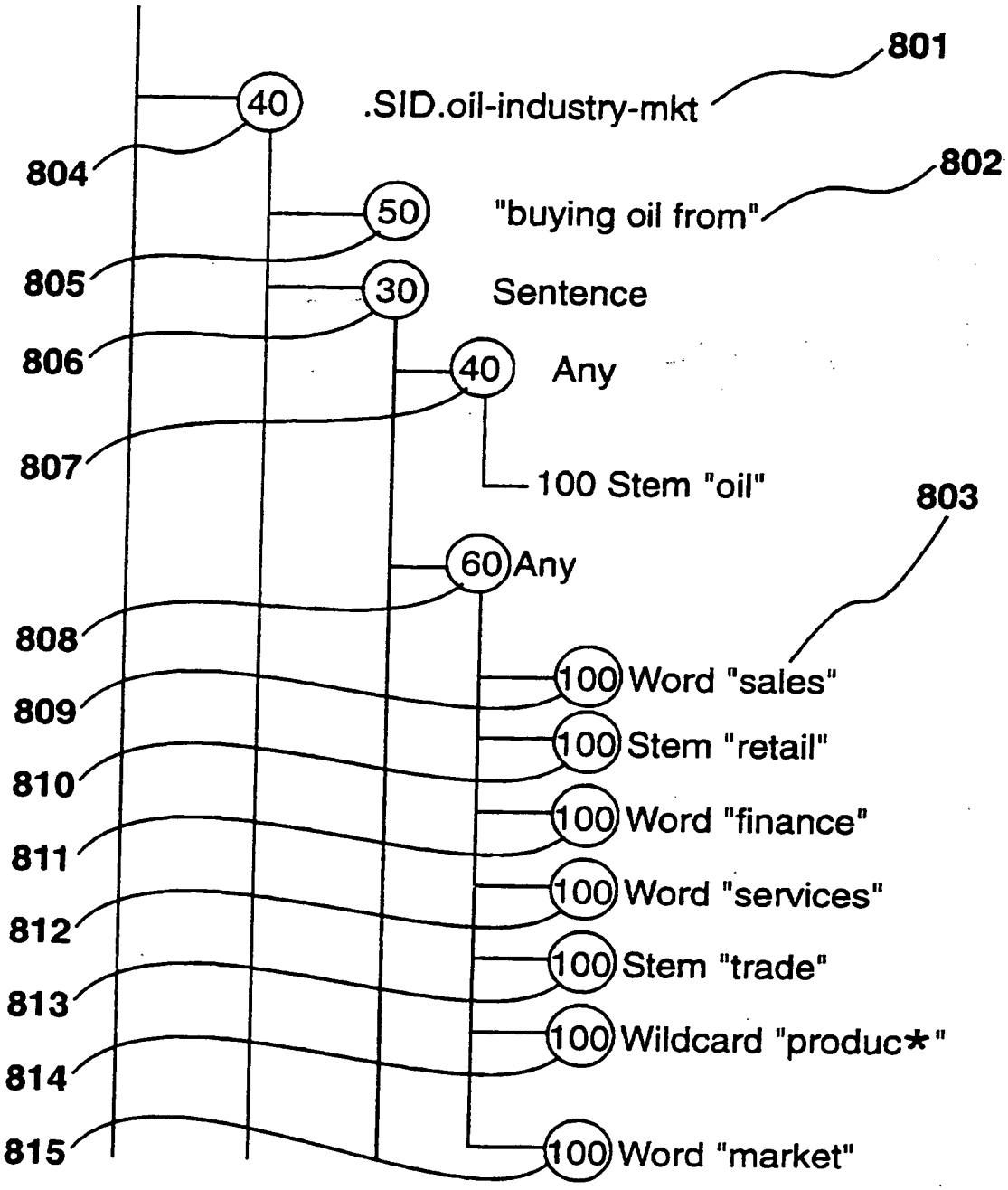
801
40
.SID.oil-industry-mkt
804
802
50
"buying oil from"
805
30
Sentence
806
40
Any
807
100 Stem "oil"
803
60
Any
808
100
Word "sales"
809
100
Stem "retail"
810
100
Word "finance"
811
100
Word "services"
812
100
Stem "trade"
813
100
Wildcard "produc*"
814
815
100
Word "market"

Figure 8

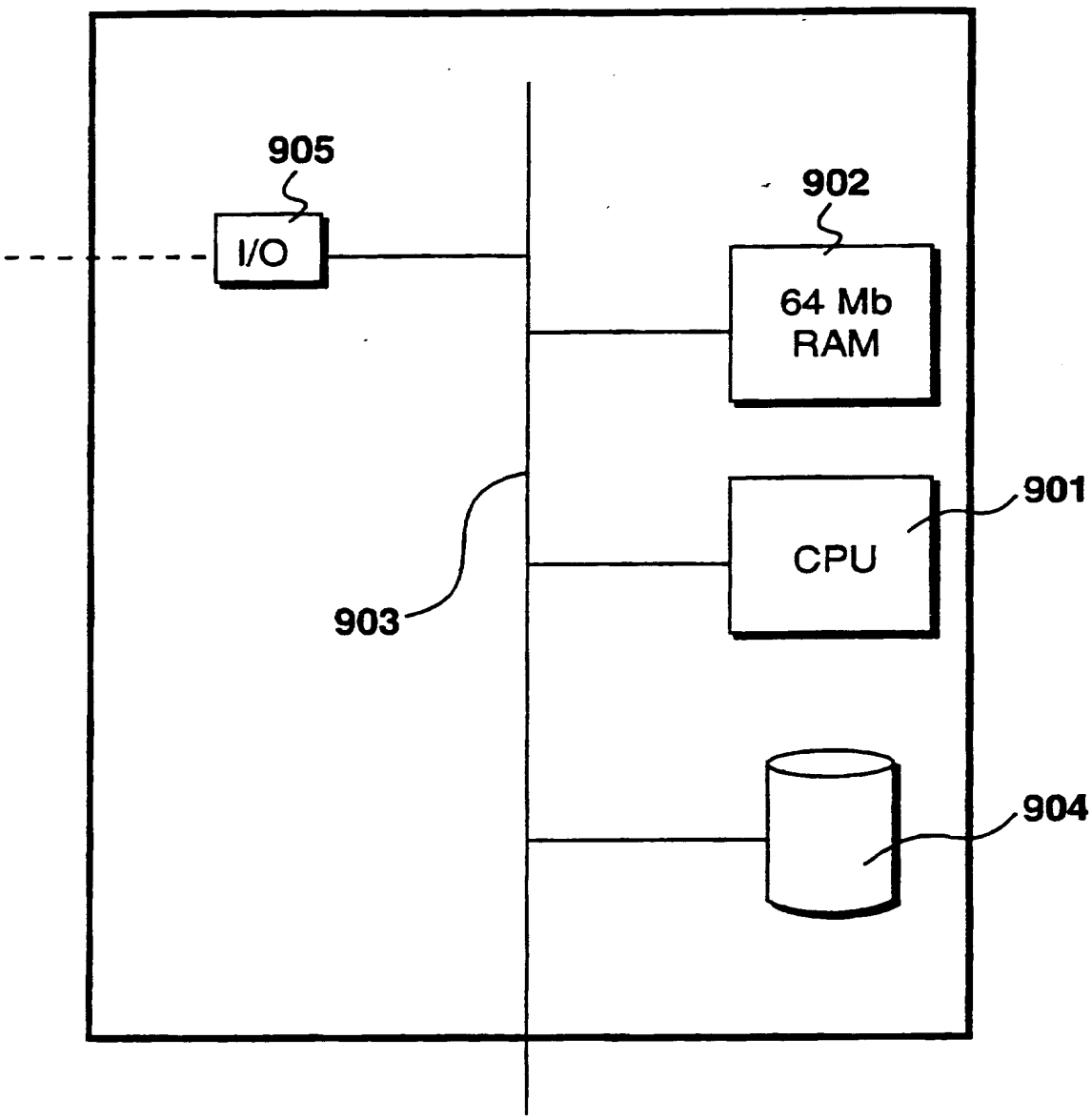

Figure 9

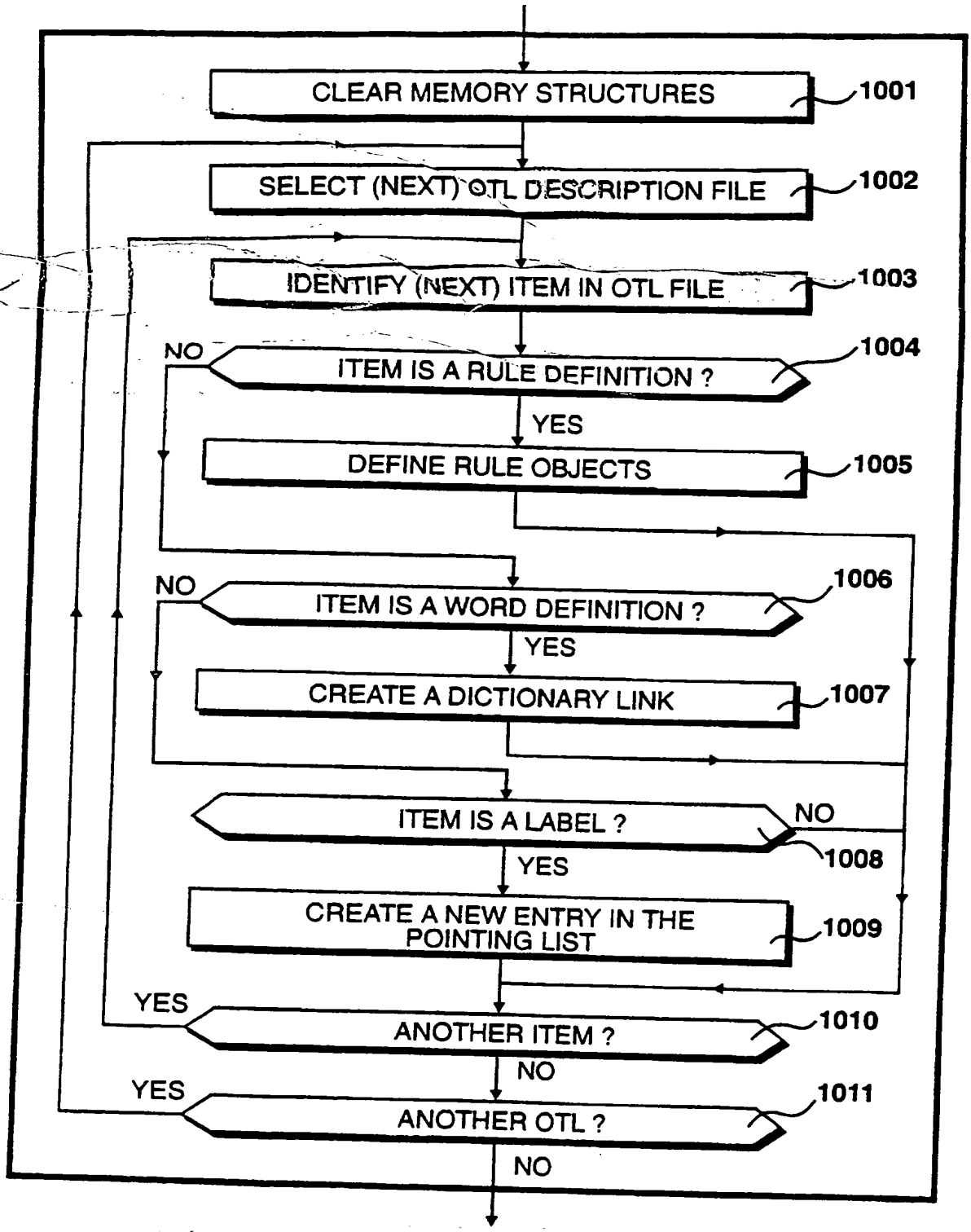
CLEAR MEMORY STRUCTURES
1001
SELECT (NEXT) OTL DESCRIPTION FILE
1002
IDENTIFY (NEXT) ITEM IN OTL FILE
1003
NO
ITEM IS A RULE DEFINITION ?
1004
YES
DEFINE RULE OBJECTS
1005
NO
ITEM IS A WORD DEFINITION ?
1006
YES
CREATE A DICTIONARY LINK
1007
ITEM IS A LABEL ?
NO
1008
YES
CREATE A NEW ENTRY IN THE POINTING LIST
1009
YES
ANOTHER ITEM ?
1010
NO
YES
ANOTHER OTL ?
1011
NO

Figure 10

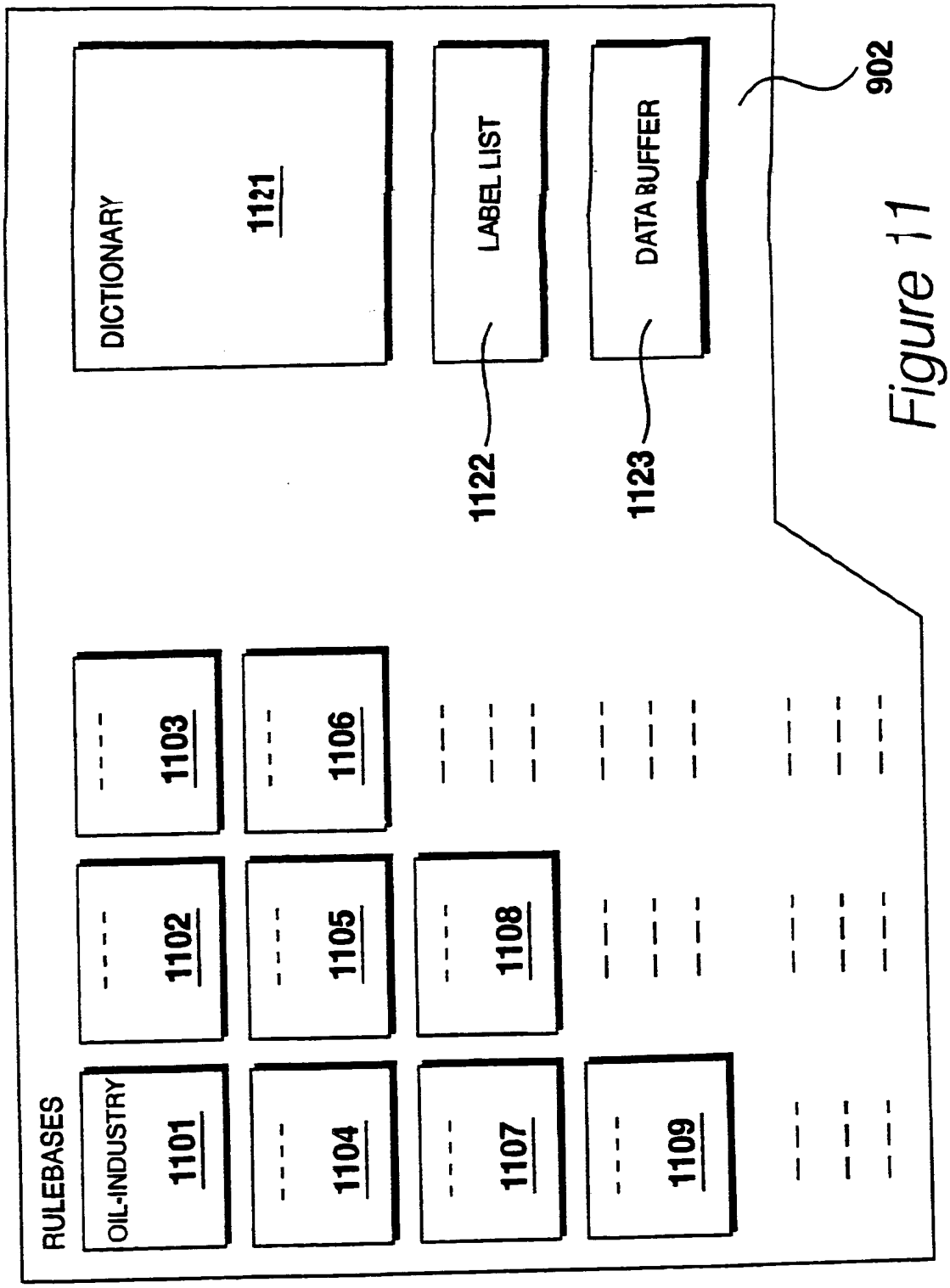
RULEBASES
OIL-INDUSTRY
1101
1102
1103
1104
1105
1106
1107
1108
1109
DICTIONARY
1121
LABEL LIST
1122
DATA BUFFER
1123
902

Figure 11

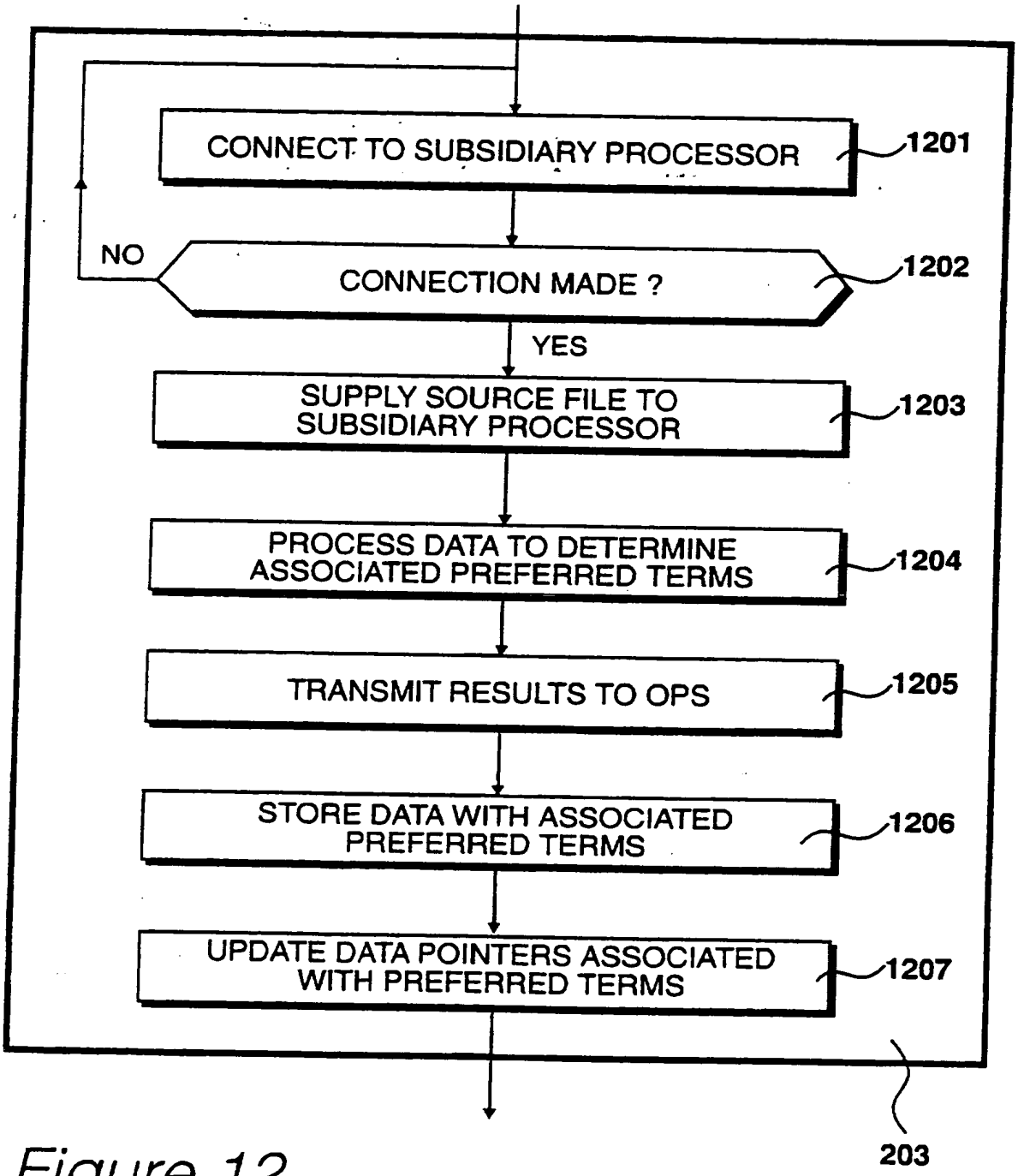
CONNECT TO SUBSIDIARY PROCESSOR
1201
NO
CONNECTION MADE ?
1202
YES
SUPPLY SOURCE FILE TO SUBSIDIARY PROCESSOR
1203
PROCESS DATA TO DETERMINE ASSOCIATED PREFERRED TERMS
1204
TRANSMIT RESULTS TO OPS
1205
STORE DATA WITH ASSOCIATED PREFERRED TERMS
1206
UPDATE DATA POINTERS ASSOCIATED WITH PREFERRED TERMS
1207
203

Figure 12

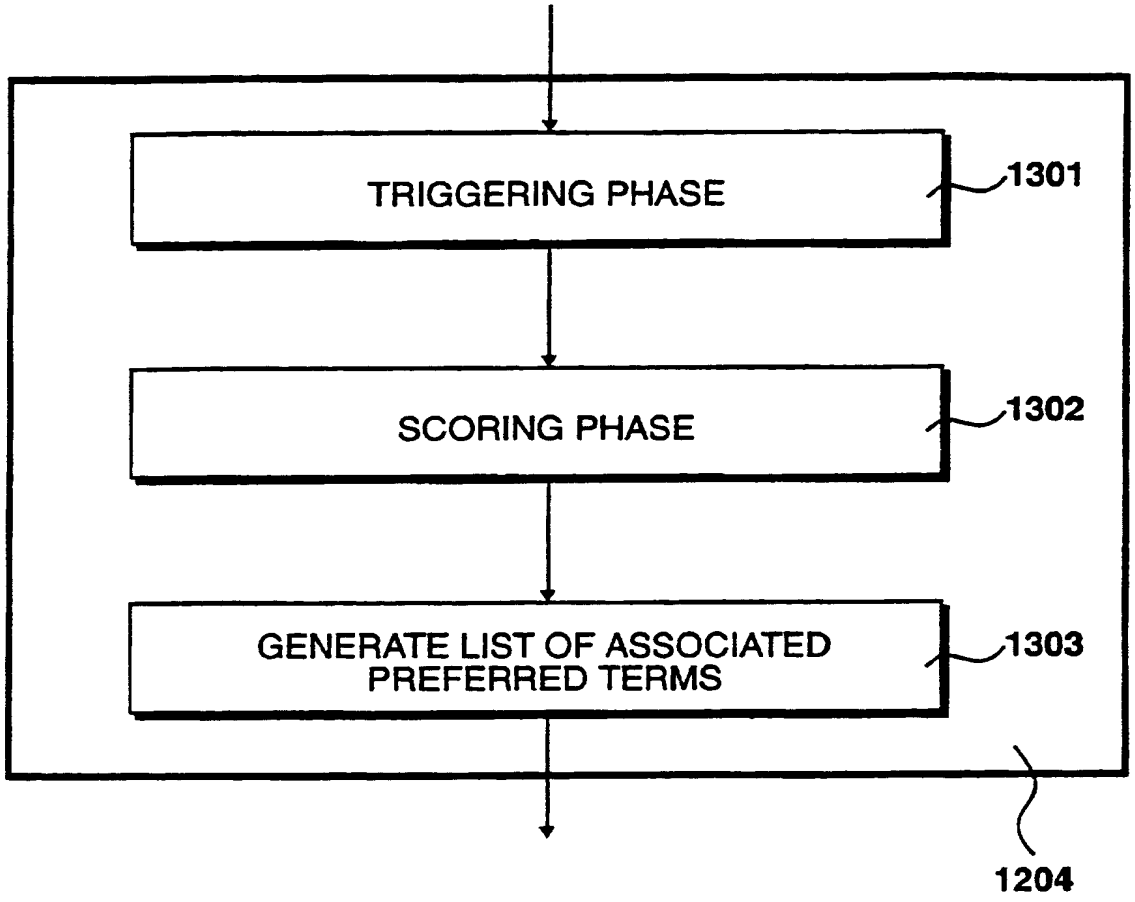
TRIGGERING PHASE
1301
SCORING PHASE
1302
GENERATE LIST OF ASSOCIATED PREFERRED TERMS
1303
1204

Figure 13

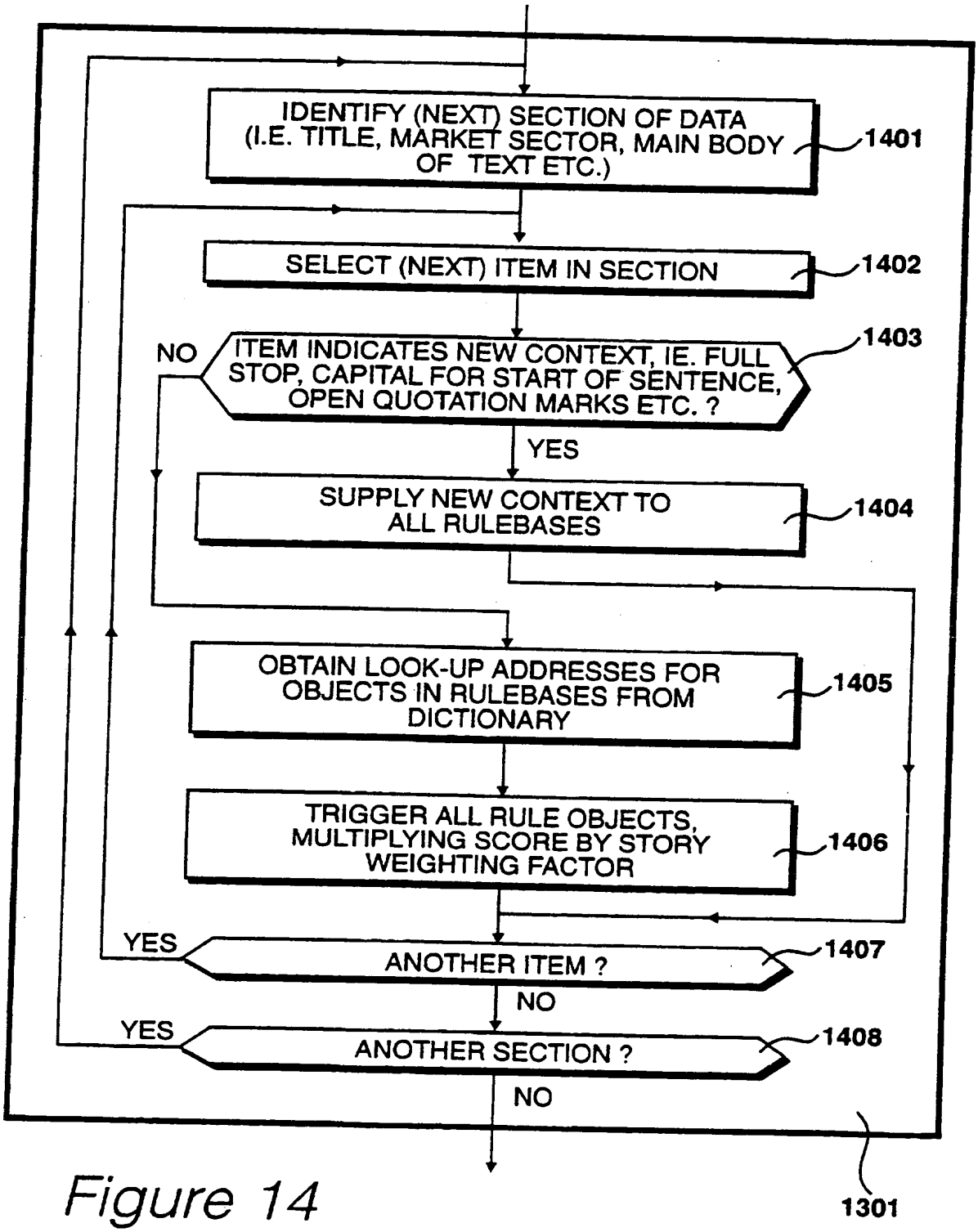
IDENTIFY (NEXT) SECTION OF DATA (I.E. TITLE, MARKET SECTOR, MAIN BODY OF TEXT ETC.)
1401
SELECT (NEXT) ITEM IN SECTION
1402
NO
ITEM INDICATES NEW CONTEXT, IE. FULL STOP, CAPITAL FOR START OF SENTENCE, OPEN QUOTATION MARKS ETC. ?
1403
YES
SUPPLY NEW CONTEXT TO ALL RULEBASES
1404
OBTAIN LOOK-UP ADDRESSES FOR OBJECTS IN RULEBASES FROM DICTIONARY
1405
TRIGGER ALL RULE OBJECTS, MULTIPLYING SCORE BY STORY WEIGHTING FACTOR
1406
YES
ANOTHER ITEM ?
1407
NO
YES
ANOTHER SECTION ?
1408
NO
1301

Figure 14

FILE WEIGHTING FACTOR. W, IS GIVEN BY :

IF S < T

$$W = \left(\frac{T - S}{T} \right) N + 1$$

ELSE

$$W = 1$$

WHERE T = THRESHOLD VALUE

S = SIZE OF CURRENT FILE

N = WEIGHTING CONSTANT, IE. 0.8

IE.

WHEN $S < 2048$, $W = \left(\frac{2048 - S}{2048} \right) 0.8 + 1$

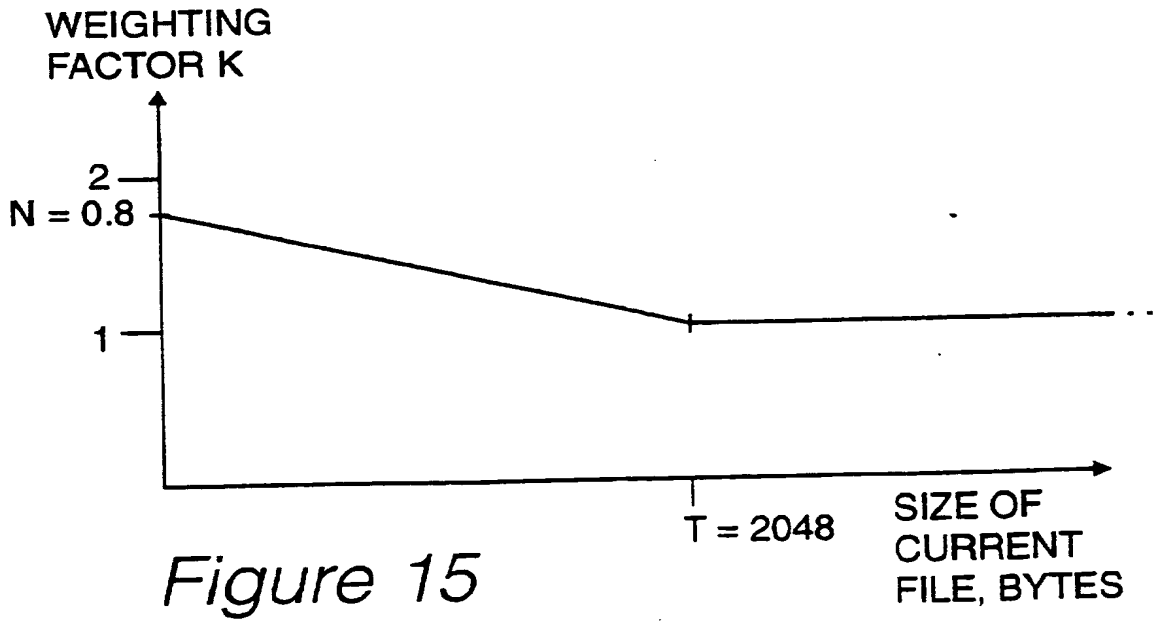

Figure 15

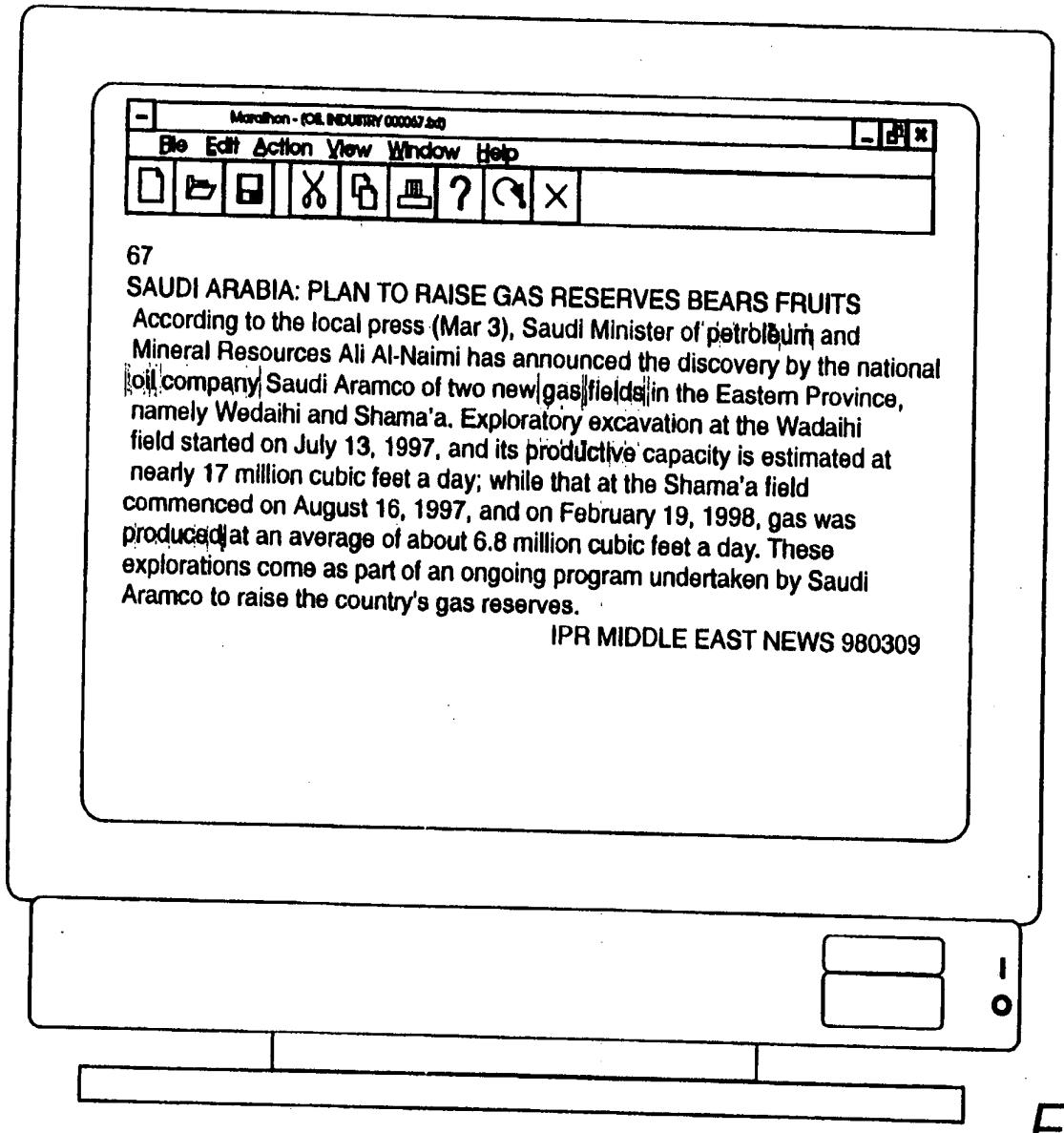
Marathon - [OIL INDUSTRY 000067.txt]
File Edit Action View Window Help
67
SAUDI ARABIA: PLAN TO RAISE GAS RESERVES BEARS FRUITS
According to the local press (Mar 3), Saudi Minister of petroleum and Mineral Resources Ali Al-Naimi has announced the discovery by the national oil company Saudi Aramco of two new gas fields in the Eastern Province, namely Wedaihi and Shama'a. Exploratory excavation at the Wadaihi field started on July 13, 1997, and its productive capacity is estimated at nearly 17 million cubic feet a day; while that at the Shama'a field commenced on August 16, 1997, and on February 19, 1998, gas was produced at an average of about 6.8 million cubic feet a day. These explorations come as part of an ongoing program undertaken by Saudi Aramco to raise the country's gas reserves.
IPR MIDDLE EAST NEWS 980309

Figure 16

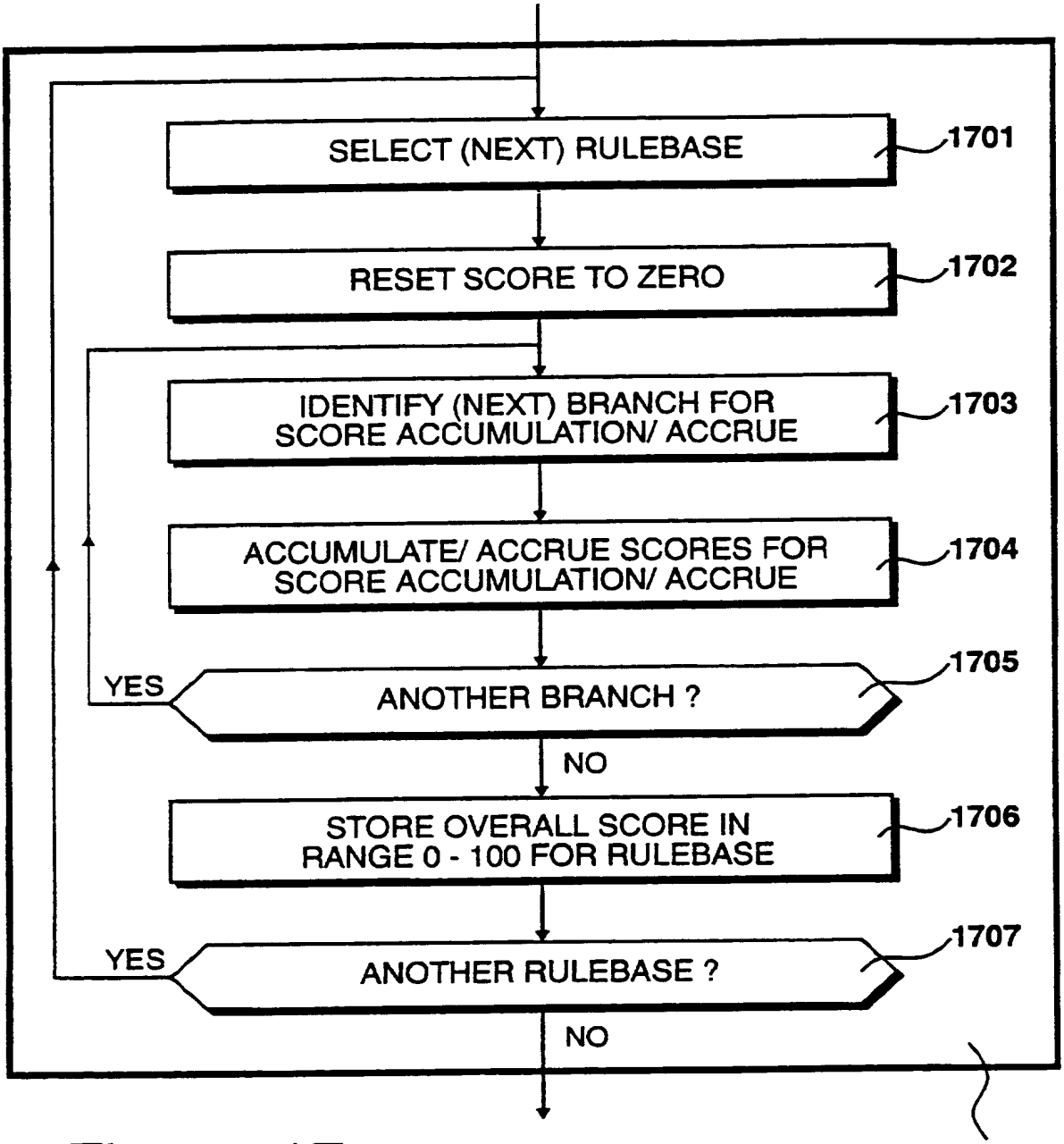
SELECT (NEXT) RULEBASE
1701
RESET SCORE TO ZERO
1702
IDENTIFY (NEXT) BRANCH FOR SCORE ACCUMULATION/ ACCRUE
1703
ACCUMULATE/ ACCRUE SCORES FOR SCORE ACCUMULATION/ ACCRUE
1704
YES
ANOTHER BRANCH ?
1705
NO
STORE OVERALL SCORE IN RANGE 0 - 100 FOR RULEBASE
1706
YES
ANOTHER RULEBASE ?
1707
NO
1302

Figure 17

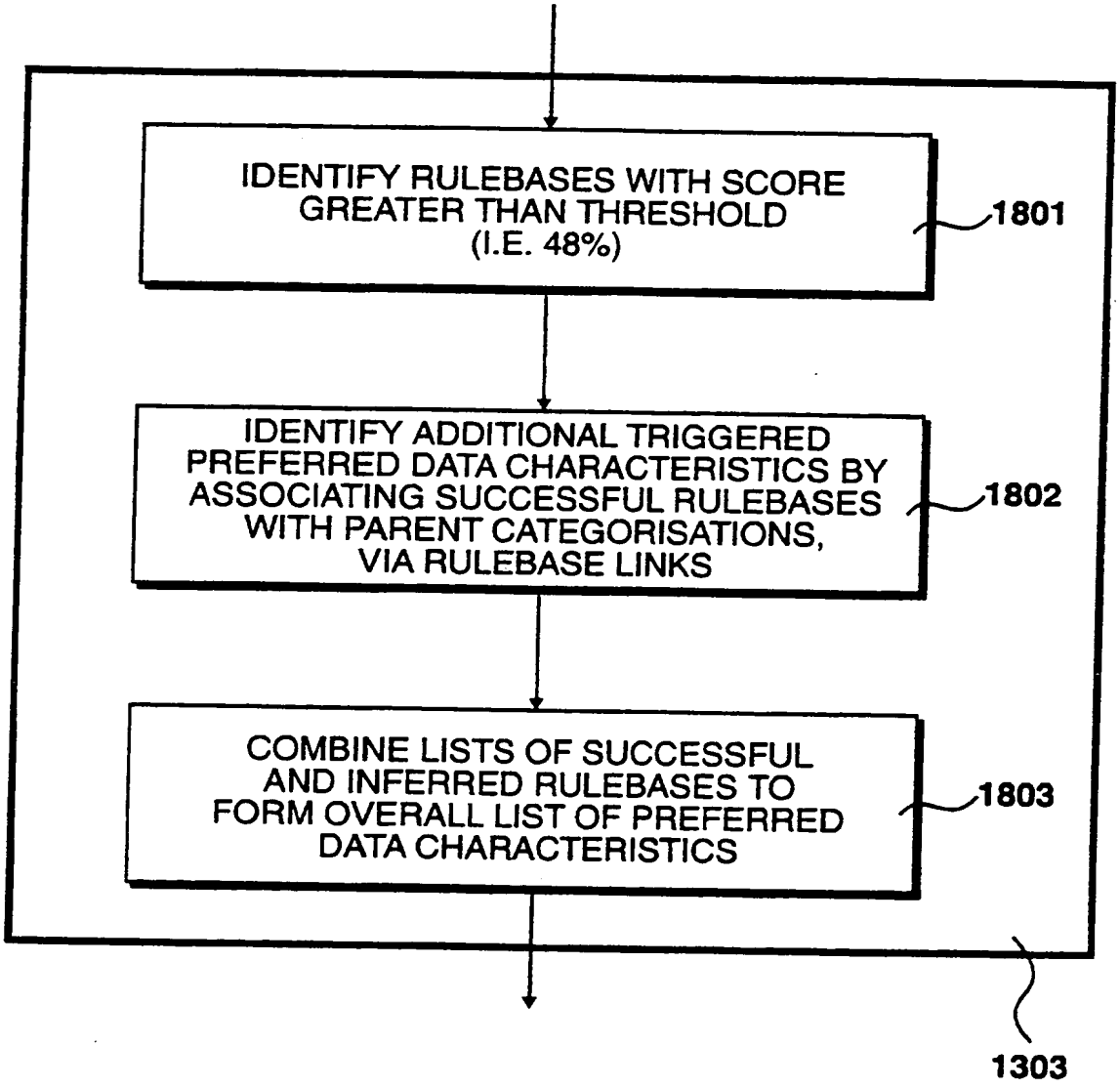
IDENTIFY RULEBASES WITH SCORE GREATER THAN THRESHOLD (I.E. 48%)
1801
IDENTIFY ADDITIONAL TRIGGERED PREFERRED DATA CHARACTERISTICS BY ASSOCIATING SUCCESSFUL RULEBASES WITH PARENT CATEGORISATIONS, VIA RULEBASE LINKS
1802
COMBINE LISTS OF SUCCESSFUL AND INFERRED RULEBASES TO FORM OVERALL LIST OF PREFERRED DATA CHARACTERISTICS
1803
1303

Figure 18

1901 PREFERRED TERM	1902 POINTER
OIL_INDUSTRY	OF8912
OIL_INSTITUTIONS	192AC3
OIL_	516321
PETROLEUM_	3200FI
⋮	⋮

Figure 19

2001 ADDRESS	2002 FILE NAME	2003 POINTER
OF8912	Oil_industry_netherland_3	OF8A20
OF8A20	Oil_ind_india_flash_	OF8193
OFA193	Petrochem_times.3.9.97	100AB1
100AB1	[END]	000000
⋮		
192AC3	BP.index_ft_uk_97	20A21B
⋮		

Figure 20

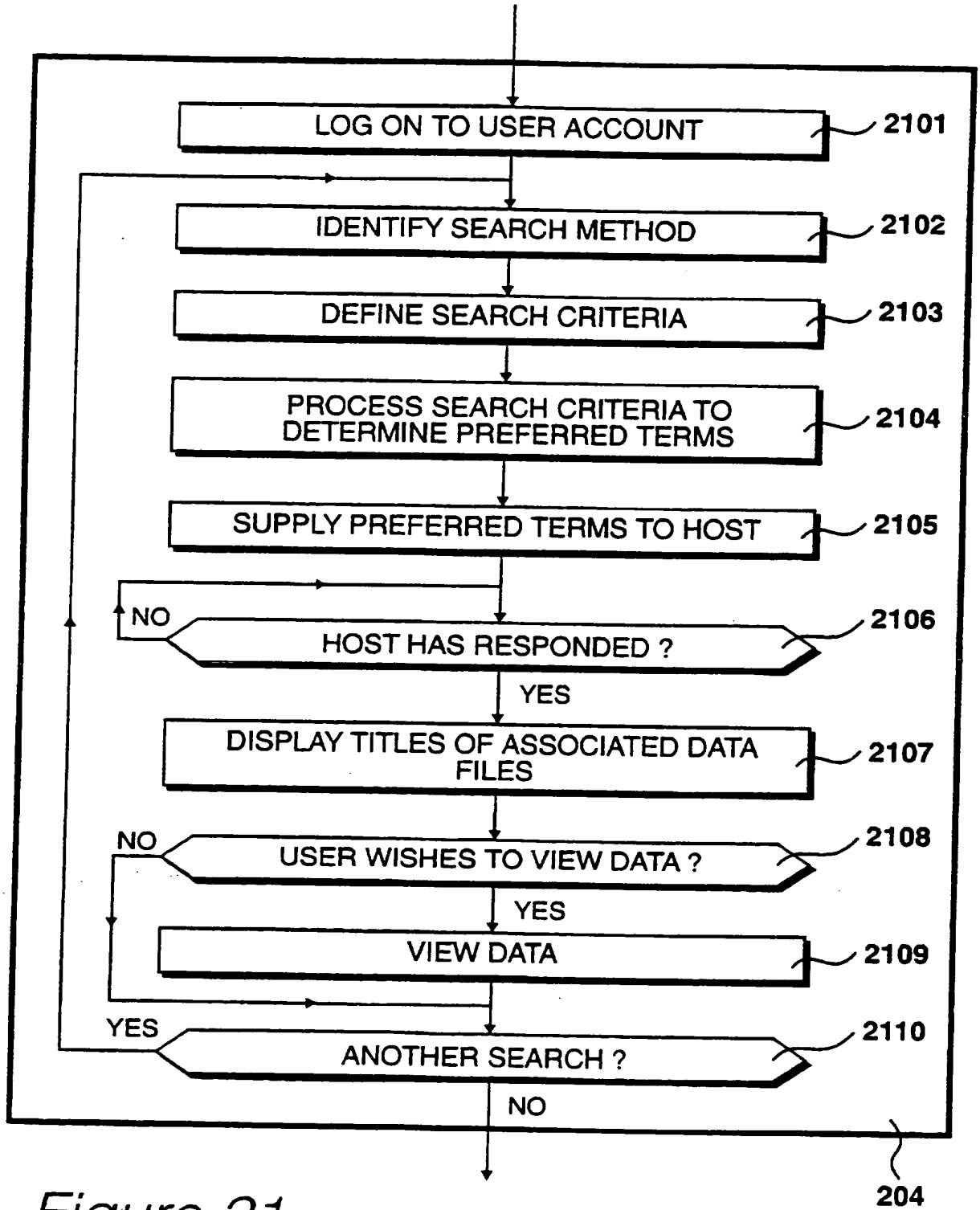
LOG ON TO USER ACCOUNT
2101
IDENTIFY SEARCH METHOD
2102
DEFINE SEARCH CRITERIA
2103
PROCESS SEARCH CRITERIA TO DETERMINE PREFERRED TERMS
2104
SUPPLY PREFERRED TERMS TO HOST
2105
NO
HOST HAS RESPONDED ?
2106
YES
DISPLAY TITLES OF ASSOCIATED DATA FILES
2107
NO
USER WISHES TO VIEW DATA ?
2108
YES
VIEW DATA
2109
YES
ANOTHER SEARCH ?
2110
NO
204

Figure 21

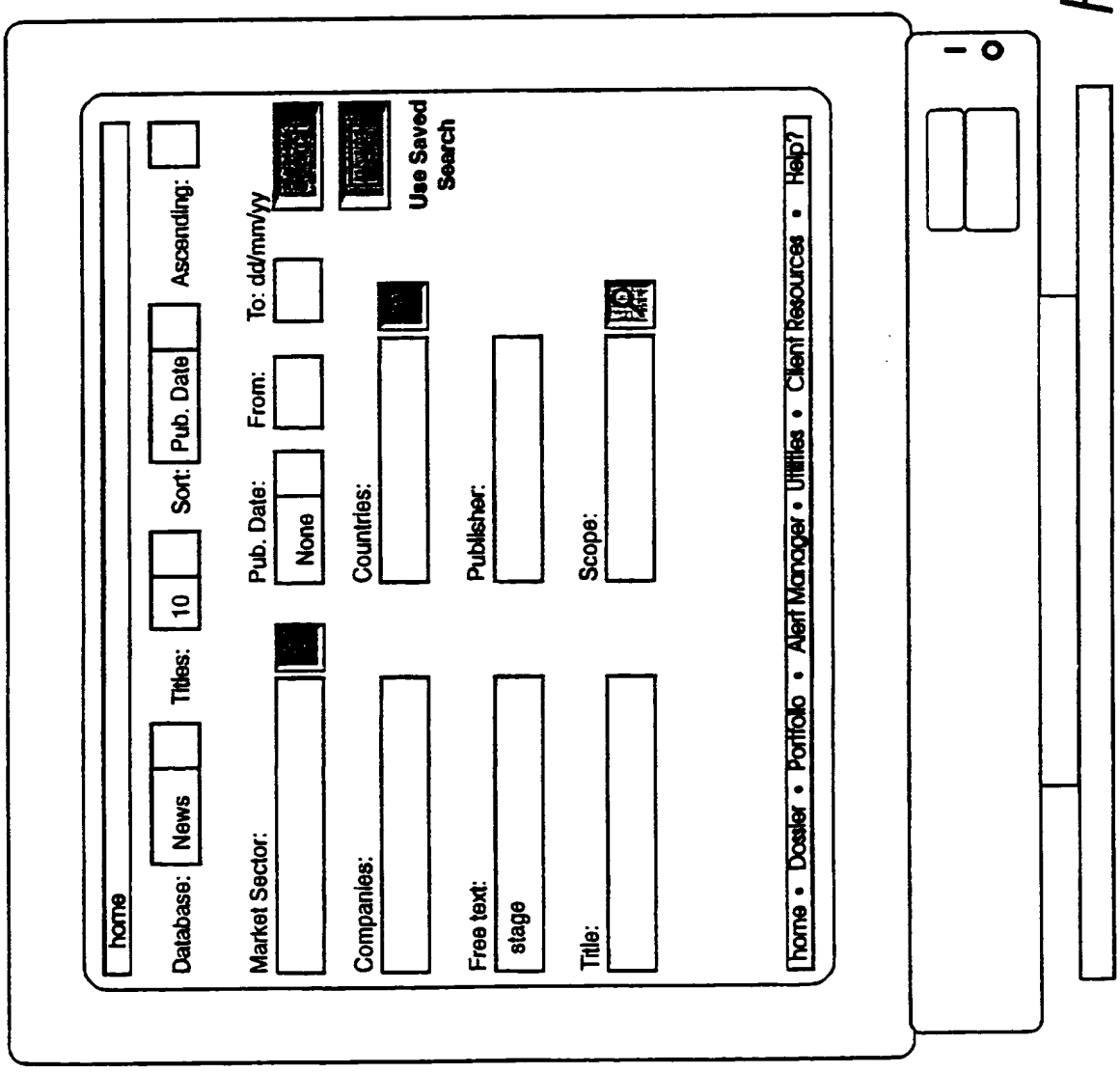
home
Database: News
Titles: 10
Sort: Pub. Date
Ascending:
Market Sector:
Pub. Date: None
From:
To: dd/mm/yy
Use Saved Search
Companies:
Countries:
Free text: stage
Publisher:
Scope:
Title:
home • Dossier • Portfolio • Alert Manager • Utilities • Client Resources • Help?

Figure 22

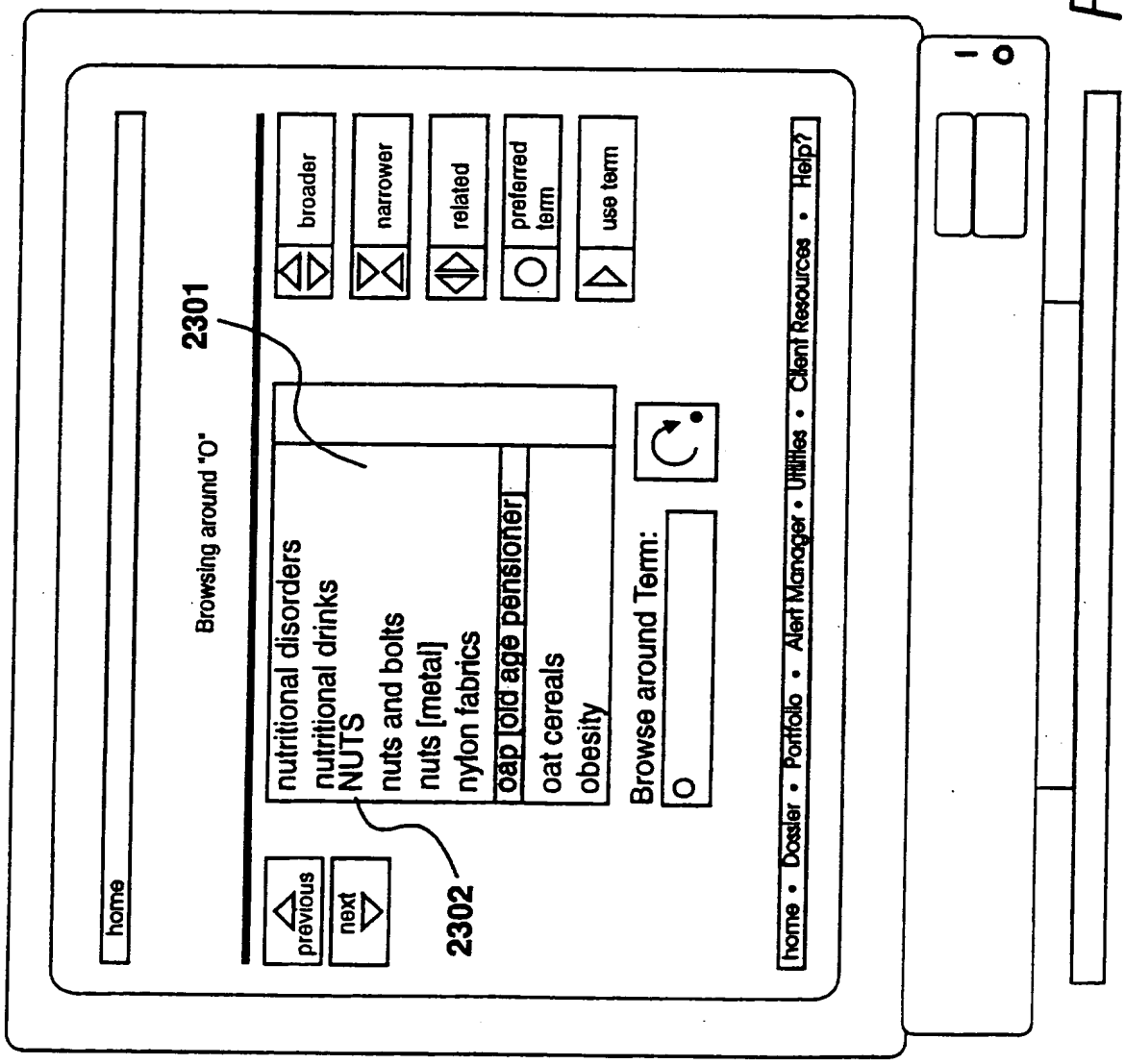
home
previous
next
Browsing around "O"
2301
2302
nutritional disorders
nutritional drinks
NUTS
nuts and bolts
nuts [metal]
nylon fabrics
oap [old age pensioner]
oat cereals
obesity
broader
narrower
related
preferred term
use term
Browse around Term:
O
home • Dossier • Portfolio • Alert Manager • Utilities • Client Resources • Help?

Figure 23

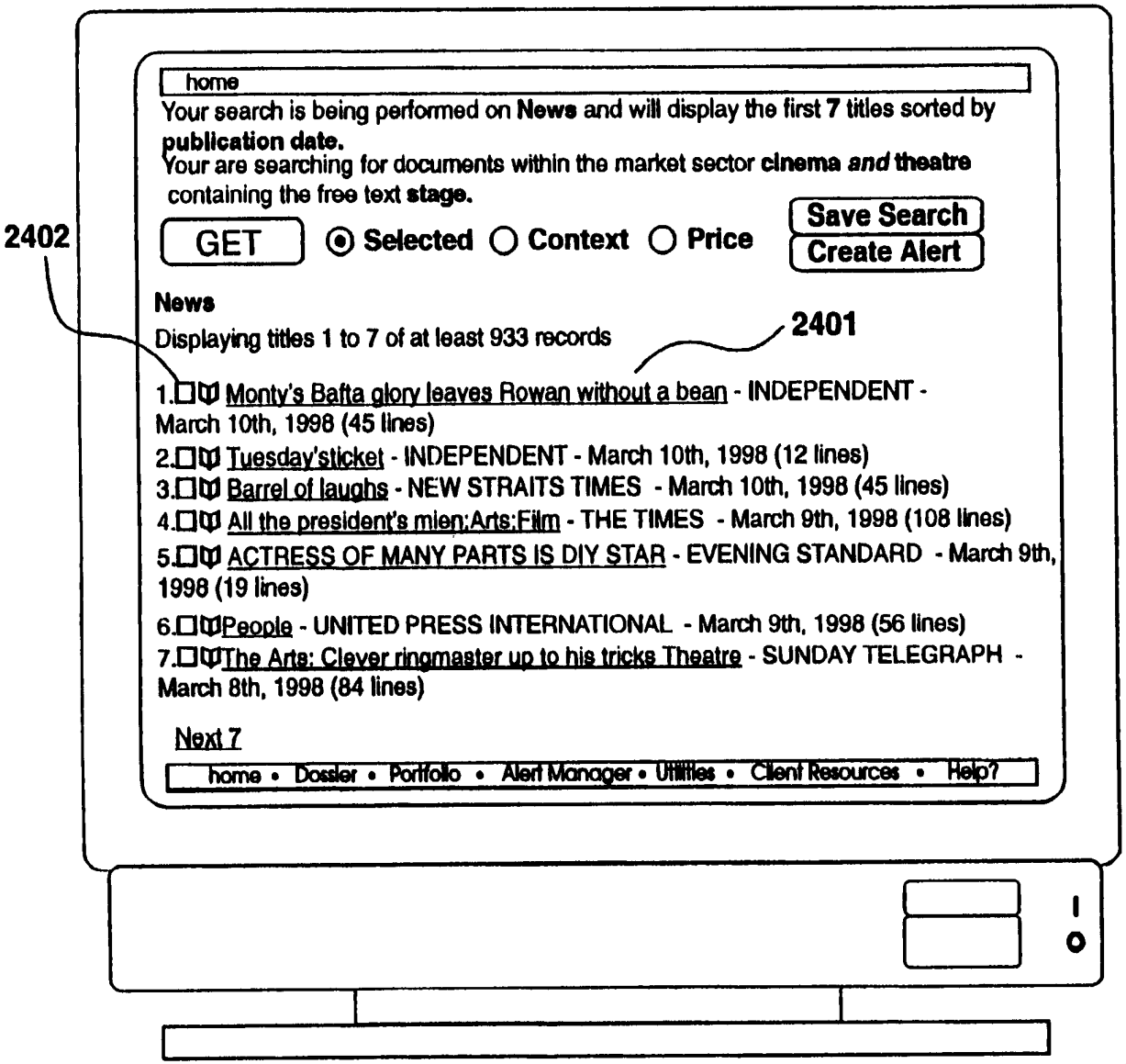
home
Your search is being performed on **News** and will display the first **7** titles sorted by **publication date.**
Your are searching for documents within the market sector **cinema *and* theatre** containing the free text **stage.**
2402
GET
Selected
Context
Price
Save Search
Create Alert
News
Displaying titles 1 to 7 of at least 933 records
2401
1. Monty's Bafta glory leaves Rowan without a bean - INDEPENDENT - March 10th, 1998 (45 lines)
2. Tuesday'sticket - INDEPENDENT - March 10th, 1998 (12 lines)
3. Barrel of laughs - NEW STRAITS TIMES - March 10th, 1998 (45 lines)
4. All the president's men:Arts:Film - THE TIMES - March 9th, 1998 (108 lines)
5. ACTRESS OF MANY PARTS IS DIY STAR - EVENING STANDARD - March 9th, 1998 (19 lines)
6. People - UNITED PRESS INTERNATIONAL - March 9th, 1998 (56 lines)
7. The Arts: Clever ringmaster up to his tricks Theatre - SUNDAY TELEGRAPH - March 8th, 1998 (84 lines)
Next 7
home • Dossier • Portfolio • Alert Manager • Utilities • Client Resources • Help?

Figure 24

ASSOCIATING FILES OF MACHINE-READABLE DATA WITH SPECIFIED INFORMATION TYPES

The present invention relates to associating files of machine-readable data with specified information types so as to enhance the availability of information from a database in which users access databases via transmission channels.

Introduction

Traditionally, database technology has been dedicated to the organisation of numerical and tabular data and it is only recently, particularly with the expansion of the Internet, that demand has grown significantly for the retrieval of text-based files. Several facilities are available on the Internet, commonly referred to as "search engines" which assist in the location of information. The majority of these operate by performing what has become known as "free text" searching, in which a user specifies words which they believe are contained within the target file as a mechanism for instructing the system to retrieve files of interest.

Problems with this technique are well known to users of the available search engines, and a simple enquiry can generate hundreds of thousands of "hits", the majority of which will tend to be totally irrelevant to the user's needs. Furthermore, other relevant files may be missed simply because they do not contain the specific chosen words.

As is well documented, a problem with the Internet is that the freedom of the Internet is also its downfall. Information is not classified before it is made available, therefore it is highly likely that even the simplest search will fail to identify relevant documentation and will take a considerable period of time to perform.

Procedures for classifying volumes of data so as to facilitate

subsequent searching are known but these classification processes often involve manual intervention, thereby making them time consuming and prone to human error. Furthermore, except in circumstances where the documentation is considered to be extremely valuable and will continue to be required over a significant period of time, the cost of performing this manual exercise cannot be justified in terms of the commercial worth of the data sources being considered. Consequently, the problem results in much data being effectively inaccessible and outside the realm of searchable knowledge.

Procedures are known for processing a data file so as to determine whether the data file should be associated with a particular information category. The known processes require a machine readable association file, also known as an outline file and usually identified by file extension .OTL. In this way, it is possible for the incoming data file to be processed with reference to one or many outline files whereupon each outline file produces a numerical score value defining an extent to which the data file is relevant to an associated category, whereafter a decision may be made on the basis of a threshold comparison.

In practical systems, thousands of such outline files would be required in order to provide a useful level of categorisation. In the present applicant's co-pending British patent application (DGC-P11-GB) a method of generating machine readable association files is described. A plurality of data files are manually selected as being examples of files which should be associated with a particular category. In addition, a plurality of files are selected manually which are considered not to be associated with a particular category. Having identified these files, the process identifies preferred term candidates from the associated files, weights these candidates with reference to files not associated with the category and applies terms to a machine readable association file by analysing the weighting values.

The resulting association files are particularly well suited to associating new data files which are of a substantially similar size to the original source data files. Similarly, association files generated by more traditional techniques still tend to be well suited to input data files of a particular size and less well suited to incoming data files of differing sizes. Thus, if a new incoming data file is smaller than the optimum file size, it is possible that relevant files will fail to be categorised given that the processing of these files will result in an inappropriately low weighting value being calculated.

Summary of The Invention

According to a first aspect of the present invention, there is provided a method of associating files of machine-readable data with information types, comprising steps of examining data elements in a file to identify occurrences of specified data types; adjusting a score in response to said identified occurrences; further adjusting said score in relation to a size of said data; and associating said information types with processed files dependent upon said score values.

In a preferred embodiment, the further adjustment is performed if the size of the data is below a predetermined threshold and a further adjustment may be performed up to a predetermined maximum weighting value.

In a preferred embodiment, the information types are expressed as preferred terms and are specified by information outlines. The information outlines may define preferred terms in a branching hierarchical structure and the adjustable values may be determined at branches of this structure. The further adjustment may include adjustment at a plurality of the branches and score values may be determined by multiplying values stored at branches throughout the hierarchy.

According to a second aspect of the present invention, there is provided apparatus for associating files of machine-readable data with information types, comprising examining means for examining data elements

in a file to identify occurrences of specified data types; adjusting means for adjusting a score in response to said identified occurrences and for adjusting said score in relation to said size of said data; and associating means for associating the information types with processed files dependent upon said score values.

In a preferred embodiment, the adjusting means is configured to further adjust the score inversely in relation to a size of said data. Preferably, means are included for determining the size of said data in terms of the volume of the stored file representing the number of characters present in the file.

Brief Description of The Drawings

Figure 1 shows a data distribution environment, including a data processing, storage and retrieval system;

Figure 2 illustrates procedures performed by the processing system shown in *Figure 1*, including a process for specifying preferred terms, a process for associating preferred terms with source files and a process for performing a search in response to a user request;

Figure 3 details the processing system shown in *Figure 1*, including subsidiary processing units;

Figure 4 details the process shown in *Figure 2* for specifying preferred terms for association with data files, including a process for the generation or modification of outline files;

Figure 5 details the step shown in *Figure 4* for the generation or modification of outline files;

Figure 6 illustrates a graphical view of an OTL file;

Figure 7 illustrates the actual OTL data for the graphical representation shown in *Figure 6*;

Figure 8 shows a diagrammatic representation of the data illustrated in *Figure 7*;

Figure 9 shows a subsidiary processor of the type identified in *Figure 3*;

Figure 10 shows operations performed by the processing unit shown in *Figure 9* in response to instructions read from the memory shown in *Figure 9*;

Figure 11 shows rule bases generated for each OTL file in response to the selection step shown in *Figure 10*;

Figure 12 details the step for associating preferred terms with source files identified in *Figure 2*, including a step for processing data to determine associated preferred terms;

Figure 13 details the step for processing data to determine associated preferred terms identified in *Figure 12*, including a triggering phase, a scoring phase and a list generation phase;

Figure 14 details the triggering phase identified in *Figure 13*;

Figure 15 details the generation of file weighting factor;

Figure 16 shows an example of a short data file;

Figure 17 details the scoring phase identified in *Figure 13*;

Figure 18 details the generation phase for a list of associated preferred terms identified in *Figure 13*;

Figure 19 illustrates a preferred term table, having pointers to a linked list;

Figure 20 details the linked list referred to in *Figure 19*;

Figure 21 details procedures performing a search in response to a user request identified in *Figure 2*;

Figure 22 illustrates a screen display generated by the requesting step shown in *Figure 21*;

Figure 23 details a screen for prompting search criteria generated by the criteria requesting step in *Figure 21*;

Figure 24 illustrates the displaying of titles of associated files generated by the procedure shown in *Figure 21*.

Detailed Description of The Preferred Embodiments

The invention will now be described by way of example only with reference to the previously identified drawings.

A data distribution environment is illustrated in *Figure 1* in which data, received from a plurality of data sources **101**, **102**, **103** is supplied to a data processing, storage and retrieval system **104**. Data sources **101** and **102** supply data directly to processing system **104** while data source **103** supplies data via a local area network **105**, thereby allowing user terminals 106 and **107** to gain direct access to their local data source **103**.

The processing system **104** provides access to a plurality of users, such as users **111**, **112**, **113**, **114**, **115**, **116** and **117**. User **111** has direct access to the processing system **104** while users **112**, **113** and **114** gain access to the processing system **104** via the Internet **118**. Users **115**, **116** and **117** exist within a more sophisticated environment in which they have access, via a local area network **119** to their own local database system **120** in addition to a connection, via an interface **121**, to the data processing system **104**.

All incoming data from data sources **101** to **103** is categorised with a key word in seven separate fields, comprising "market sector", "location", "company name", "publisher", "publication date" and "scope". Users, such as users **112** to **117** may specify almost any term as the basis for a search and are then prompted by an equivalent word or phrase which constitutes more preferred search parameters. For example, a user may specify a search word such as "confectionery" and the system will prompt the user to consider narrower terms such as "chocolate" along with related terms such as "cakes" or "desserts", or broader terms such as "food". From a simple request, a user is given an option of focusing further or of taking a broader overview of the subject under consideration.

The scope of an article refers to the context in which the document or article was written. For example, the scope field may consider questions as to whether the article concerns "mergers and acquisitions" or "seasonal trends" et cetera. Such terms are useful in gathering related information from a wide variety of industries and markets and may prove invaluable for particular applications.

The same criteria used for indexing are offered for search purposes and the same indexing terms are used for all documents across a range of specific databases. An overview of procedures performed by the data processing system **104** is illustrated in *Figure 2*. At step **201** preferred terms for association with data files are specified. This step is essentially performed as an "off-line" process; establishing the environment for allowing source data to be processed as it is received from sources.

Steps **202**, **203** and **204** represent on-line procedures after the preferred terms have been specified at step **201**. At step **202** the processing system **104** receives data from sources such as sources **101**, **102** and **103**. The source data may be transmitted using different protocols, formats and standards therefore the processing system performs a standardisation process so that the data may be stored locally at the data processing system using standardised formats.

At step **203** the data is processed so as to enhance a user's ability to identify information of interest. Files of machine-readable data received from the sources are associated with specific preferred terms which may be considered as defining particular information types. A file is considered and individual data elements, usually in the form of natural language words, are examined to identify occurrences of specified data types. The purpose of this association is to identify files of data which are of interest in relation to particular topics. This enables a user to organise a search which should result in useful information being supplied to said user, with reference to said topics and defined terms, from an extremely large database of stored data

files. In this way, the technical procedures performed by association step **203** significantly enhances the overall functionality of the system and provides an industrially applicable approach to allowing highly focused sets of information to be supplied to a user in preference to large volumes of data; much of which will tend to be totally irrelevant.

In order to achieve this, the files of data are considered and are given a score representing a numerical value as to their relevance with respect to the predefined topics. Scores are adjusted in response to the number of identified occurrences of a specified data type. Furthermore, these scores are also adjusted in relation to the size of the data contained within the file. In particular, occurrences of data types in relatively small files are given a higher weighting with occurrences in larger files being given a lower weighting. Thus, the adjustment of scores is related inversely to the actual size of the data file. Thereafter, a threshold for the scoring values may be set and information types are associated with particular files dependent upon whether particular value scores fall on one side or on the other side of this threshold.

At step **204** a search is performed, in response to preferred terms identified by a user such that information of interest may be identified within the data stored by the data processing system **104** and transmitted to user terminals, such as terminal **111**, over transmission channels as illustrated in *Figure 1*.

Processing system **104** is detailed in *Figure 3*. Data signals from data sources **101** to **103** are supplied to input interfaces **301** via data input lines **302**. Similarly, output data signals are supplied to users **111** to **117** via an output interface **303** and output wires **304**. Input interface **301** and output interface **303** communicate with a central processing system **305** based on DEC Alpha integrated circuitry. The central processing system **305** also communicates with other processing systems in a distributed processing architecture. Processing system **104** includes eight Intel chip based processing systems **311** to **318**, each implementing instructions under the

control of conventional operating systems such as Windows NT.

An operator communicates with the processing system **104** by means of an operator terminal, having a visual display unit **321** and a manually operable keyboard **322**. Data files received from sources **101** to **103** are written to bulk storage devices **323** in the form of large magnetic disk arrays. Data files are written to disk arrays **323** after these files have been associated with preferred terms, as illustrated at step **203**. These association processes are performed by the subsidiary processors **311** to **318** and the central processing system **305** is mainly concerned with the switching and transferring of data between the interface circuits **301**, **303** and the disk arrays **323**.

The central processing system **305** communicates with the subsidiary processors **311** to **318** via an Ethernet connection **324** and processing requirements are distributed between processors **311** to **318**. Having addressed a subsidiary processor **311** to **318** the transferring of data to an addressed processor is performed. Each individual incoming data file is supplied exclusively to one of the subsidiary processors. The selected subsidiary processor is then responsible for performing the association process, to identify preferred terms relevant to that particular data file. Thereafter, the associated data file is returned to the central processing system **305**, over connection **324** and the central processing system **305** is then responsible for writing the associated data file to the disk array **323**. In this way, it is possible to scale the degree of processing capacity provided by system **104** in dependence upon the volume of data files to be processed in this way. The central processing system **305** also maintains a table of preferred terms, pointing to particular data files which have been identified as relevant to said preferred terms.

Process **201** for specifying preferred terms for association with data files is detailed in *Figure 4*. At step **401** a preferred term is selected and at step **402** an outline (OTL) file is generated or modified. At step **403** a

question is asked as to whether another term is to be processed and when answered in the affirmative control is returned to step **401**, allowing the next term to be processed at step **402**. Eventually, all of the terms will have been processed resulting in appropriate generations or modifications to their related outline files. Consequently, the question asked at step **403** is answered in the negative whereafter at step **404** data structures are initialised by parsing the OTL files generated at step **402**.

Step **402** for the generation or modification of outline files is detailed in *Figure 5*. At step **501** a visual OTL editor is opened resulting in the editor's visual interface being displayed on VDU 321. At step **502** a question is asked as to whether an existing file is to be loaded for modification and if answered in the negative a new OTL file is created at step **503**. If the question asked at step **502** is answered in the affirmative, step **503** is bypassed and at step **504** modifications or additions are made to the OTL definition. At step **505** the OTL modifications created at step **504** are tested on a sample of test data and at step **506** a question is asked as to whether another modification is to be made. When answered in the affirmative, control is returned to step **504** resulting in further modifications or additions being made to the OTL definitions. When answered in the negative at step **506**, the new or modified OTL file is saved at step **507**.

When performing modifications or additions at step **504**, a graphical representation of the OTL file data is presented to an operator via the visual display unit **321**. An example of a display of this type is illustrated in *Figure 6*, representing a graphical illustration of a specific OTL file.

The OTL file stores definitions in an hierarchical tree structure and this structure is represented in the graphical view as shown in *Figure 6*. A representation of the tree may be contracted or expanded and the possibility of expanding a particular branch is identified by a plus sign on a particular line, as shown at **601**. Similarly, when a particular branch has been fully expanded, the line is identified by a minus sign as shown at **602**. Definitions

within the file consist of rules, words and labels. The labels allow relationships to be defined between various parts of the file and between individual files themselves. The words identify specific words within an input file of interest and the rules define how and what weights are to be attributed to these words. Each rule line includes, at its beginning, a weight value **603** representing the score that will be attributed when a particular rule condition is met. Rules may also have leaves and the rule defines the way in which scores generated from leaves are combined.

OTL file data represented graphically in the form shown in *Figure 6* is actually stored in a data file having a format of the type shown in *Figure 7*. The actual data file shown in *Figure 7* corresponds to the data display in *Figure 6* but in *Figure 7* all of the data, some of which has been rolled up in *Figure 6*, is present. The data contained within the file shown in *Figure 7* is manipulated interactively by an operator in response to the graphical interface displayed as illustrated in *Figure 6*. Score values **603** are also identified in the data file shown in *Figure 7*.

Displayed line **601** in *Figure 6* is generated from line **701** of the actual stored data. The syntax of the language used for recording the data, as illustrated in *Figure 7*, may vary and the example shown is specific to this particular application. However, the underlying functionality of the language may be considered with reference to the diagrammatic representation shown in *Figure 8*.

Purely to provide a specific example, this particular outline file is concerned with the topic of the oil industry and therefore the purpose of the OTL file is to identify words and phrases within an input file so as to provide an indication as to how relevant that input data is to users having an interest in the oil industry. Thus, the purpose of procedures exploiting these OTL files is to generate evidence showing that a particular data file conveys information which may be of interest to those studying the oil industry.

The outlines analyse data files in order to produce numerical evidence as to the relevance of a particular file with relation to a particular topic. The OTL definitions and structures are determined empirically and would be modified and upgraded over a period of time. The system does more than merely register the existence of a particular word item by placing the word items within an interacting structure; the nature of which is illustrated in *Figure 8*. The particular entry, given label "oil-industry-mkt" relates to marketing aspects of the oil industry and as such can contribute to an overall score as to the pertinence of incoming data to this particular topic. The first line **801** shows that this particular contribution may provide a total score of forty percent. This total of forty percent is then subdivided such that at line **802** the presence of the phase "buying oil from" has a score of fifty percent. Thus, the total contribution made the presence of this phrase consists of fifty percent of forty percent, ie a total of twenty percent being made to the total contribution. Similarly, as shown at line **803** and below, particular words may be identified which result in contributions of sixty percent of thirty percent of forty percent. Thus, a complete OTL file is structured in this way with particular words and phrases making contributions to an overall score value. These words and phrases may also be specified in the rules as making single contributions or being allowed to accrue.

Examples of score value **603** are illustrated in *Figure 8* at **804** to **815**. The hierarchical structure in *Figure 9* consists of a plurality of branches with lowest level entries being considered as leaves. Each leaf has a score value associated with it and values **809** to **815** are leaf score values. Above these, branch score values exist such that score values **807** and **808** exist at the lowest level of branching with score values **805** and **806** being at the next level of branching further up each connected to the highest level of branching illustrated by score value **804**.

A total score value for a particular occurrence, detected at any level within the hierarchical tree, results in a final score contribution derived from

the product of the score value assigned to that particular level with all score levels identified while ascending the tree structure back to its root.

The score contributions are produced and possibly accumulated when occurrences of the specified elements are identified. This provides a numerical weight to assess whether a file being processed should be associated with a particular information type. The present invention, as implemented within the preferred embodiment, further adjusts these score weight in relation to the size of the overall data file. In particular, it is the branch score values (**804** to **808**) which are modified in preference to leaf values (**809** to **815**).

Subsidiary processor **311** is detailed in *Figure 9*. The processor includes an Intel Pentium processing unit **901** connected to sixty-four megabytes of randomly accessible memory **902** via a PCI bus **903**. In addition, a local disk drive **904** and an interface circuit **905** are connected to bus **903**. Interface circuit **905** communicates with the TCP/IP network **324**.

Random access memory **902** stores instructions executable by the processing unit **901**, in addition to storing input data files received from the data sources **101** to **103** and intermediate data. Operations performed on processing unit **901**, in response to instructions read from memory **902** are identified in *Figure 10*.

At step **1001** temporary memory structures are cleared and at step **1002** an OTL description file is selected. At step **1003** an item in the OTL file is identified and at step **1004** a question is asked as to whether the item selected at step **1003** is a rule definition. If this question is answered in the affirmative, a rule object is defined at step **1005**. Alternatively, if the question asked at step **1004** is answered in the negative, to the effect that the item is not a rule definition, a question is asked at step **1006** as to whether the item is a word definition. If this question is answered in the affirmative, a dictionary link is created at step **1004**.

At step **1008** a question is asked as to whether the item is a label and when answered in the affirmative a new entry is created in a label list, whereafter at step **1010** a question is asked as to whether another item is present. After executing step **1005** or after executing step **1007**, control is directed to step **1010**.

When the question asked at step **1010** is answered in the affirmative, to the effect that another item is present, control is returned to step **1003** and the next item is identified in the OTL file. Eventually, all of the items will have been identified resulting in the question asked at step **1010** being answered in the negative. Thereafter, at step **1011** a question is asked as to whether another OTL file is present and when answered in the affirmative control is returned to step **1002** allowing the next OTL description file to be selected. Thus, this process continues until all of the OTL files have been considered resulting in the question asked at step **1011** being answered in the negative.

For each OTL file considered, by being selected at step **1002**, a rule base is generated and a plurality of such rule bases are illustrated in *Figure 11*. Thus, a first OTL file processed in accordance with the procedures shown in *Figure 10* results in the generation of a first rule base **1101**. Similarly, further iterations of the procedures shown in *Figure 7* result in the generation of rule bases **1102** to **1109**. Typicaiiy, for a specific installation, in the order of three thousand rule bases would be generated by execution of the procedures illustrated in *Figure 10*.

Rule bases **1101** to **1109** are stored in memory **902**, which also provides storage space for a dictionary **1121**, a label list **1122** and a data buffer **1123**. The dictionary stores a list of words which have importance in any of the stored rule bases. Associated with each word in the dictionary there is at least one pointer and possibly many pointers to specific entries in specific rule bases **1101** to **1109**. Thus, the words identified at **803** in *Figure 8* would all be included in dictionary **1121**. Entries within the dictionary **1121** are implemented upon execution of step **1007** in *Figure 10*. Similarly,

execution of step **1009**, creating a new entry in the label list, allows a label to relate to rules that are elsewhere in the tree structure. Step **203**, as shown in *Figure 2* for associating preferred terms with source files, is detailed in *Figure 12*. At step **1201** central processor **305** obtains access to one of the subsidiary processors **311** to **318**. The central processor then expects to receive authorisation so that communication may be effected with one of the subsidiary processors, after a connection has been established, the source file is supplied to the selected subsidiary processor at step **1203** and at step **1204** the data is processed to determine associated preferred terms.

After performing the processing at step **1204**, the results are transmitted back to the central processing system at step **1205** and at step **1206** data with associated preferred terms is stored and data pointers associated with the preferred data terms are updated at step **1207**.

Step **1204** for the processing of data to determine associated preferred terms is detailed in *Figure 13*. The overall processing is broken down into three major phases, consisting of a triggering phase at **1301**, followed by a scoring phase at **1302** followed finally by a list generation phase at step **1303**.

Triggering phase **1301** is detailed in *Figure 14*. At step **1401** a section of the data, such as its title, market sector or main body of text, is identified and at step **1402** an item of the identified section is selected. At step **1403** a question is asked as to whether the item indicates a new context, which may be considered as a grammatical marker in the form of a full stop, capital, start of a sentence or quotation marks et cetera. When answered in the affirmative new context information is supplied to all rule bases **1101** to **1109** at step **1404** and control is then directed to step **1407**.

If the question asked at step **1403** is answered in the negative, step **1404** is bypassed and a look-up address is obtained for rule objects in rule bases from the dictionary at step **1405**. Thereafter, at step **1406** all addressed objects are triggered and a multiplication of scores is effected by a

score weighting factor. Thereafter, at step **1407** a question is asked as to whether another item is present and when answered in the affirmative control is returned to step **1402**.

Eventually, all of the items for a selected section will have been considered resulting in the question asked at step **1407** being answered in the negative. Thereafter, at step **1408** a question is asked as to whether another section is to be considered and when answered in the affirmative control is returned to step **1401**.

At step **1401** the next section is identified and steps **1402** to **1408** are repeated. Eventually, all of the sections will have been considered and the question asked at step **1408** will be answered in the negative.

As shown in *Figure 8*, each lowest level leaf of the hierarchical tree has a numerical value associated with the identification of a particular item, as identified generally at **803**. If the amount of data contained within a particular file is less than what would generally be accepted, scores are further adjusted in relation to this size so as to improve the association of files with particular information types. This further adjustment is performed at the lowest leaf level (an example of this being level **803** in *Figure 8*) and these leaf values are multiplied by a file weighting factor, derived from the size of the file, which is triggered at step **1406** as shown in *Figure 14*.

The generation of a file weighting factor W is illustrated in *Figure 15*. Experiments have shown that below a certain file size, the irregular distribution of evidential terms, referred to as granularity, can effect the accuracy of the association process and therefore requires compensation. In order to provide compensation, a threshold value is identified, below which compensation is required, and recorded by variable T. A typical example for T would be two thousand and forty-eight bytes. The size of the current file is recorded in variable S and a weighting constant, such as zero point eight (0.8), is stored by variable N. From these values, weighting factor W is derived as follows. If the size S of the current file is smaller than threshold T,

weighting factor W is calculated by subtracting the size S of the current file from the threshold T and dividing this by the threshold T. This partial result is then multiplied by the weighting constant and added to unity. Alternatively, if the size of the current file is not smaller than the threshold T, the weighting factor is set to unity.

The threshold value T represents the minimum file size that can be indexed reliably and files smaller than this value, when detected, result in further adjustments being made to the identification process so as to compensate for this problem. The threshold value could be calculated on an on-going basis so as to provide, for example, the average of the previous one thousand files considered. However, given the inherent unpredictability of file size distribution, it is better to specify a threshold value and, in some circumstances, it may be preferable to modify the threshold value in order to achieve a particular result. Furthermore, the weighting constant may also be modified in response to particular data types and experiments have shown that by optimising the weighting constant and by optimising the threshold value it should be possible to obtain a ten percent improvement in terms of the number of files that are selected correctly for a particular category.

The relationship between weighting factor and file size is illustrated graphically in *Figure 15* and it can be seen that, as the file size falls below the threshold value, the weighting factor is linearly increased up to the value of the weighting constant. Alternatively, a more sophisticated procedure may be achieved, having a non-linear relationship, by using interpolated look-up tables.

An example of a short data file is illustrated in *Figure 16*. The file does not contain many words therefore it is unlikely that enough words would be identified so as to provide evidence for the data file to be included within a particular category identified by its associated preferred terms. Figure 16 also shows particular words and phrases that have been highlighted, so as to identify them as having being caught by a particular rule base. Thus, a

decision then has to be made as to whether these highlighted words provide sufficient evidence for the data file to be included. The present procedures, as illustrated in *Figure 15*, would increase the likelihood of a file of this type being included correctly and as being pointed to by a particular preferred term.

Scoring phase **1302** is detailed in *Figure 17*. At step **1701** a rule base is selected and at step **1702** a score variable is re-set to zero. At step **1703** a branch is identified for score accumulation/accrue and at step **1704** scores are accumulated or accrued from triggered rules attached to the branch. At step **1705** a question is asked as to whether another branch is to be considered and when answered in the affirmative control is returned to step **1703**. A next branch is selected at step **1703** with procedure **1704** being repeated. Eventually all of the branches will have been considered resulting in the question asked at step **1705** being answered in the negative.

At step **1706** an overall score in the range of zero to one hundred is stored for the rule base and at step **1707** a question is asked as to whether another rule base is present. When answered in the affirmative, control is returned to step **1701** and steps **1701** to **1707** are repeated. Eventually, all of the rule bases will have been considered and the question asked at step **1707** will be answered in the negative.

The operations illustrated in *Figure 17* may be considered with reference to the illustration of the structure in *Figure 8*. Thus if any of the defined words at **803** are identified within the file a provisional score of one hundred will be allocated. However, the process as shown in *Figure 17*, must then ascend up the branches so that any scores lower down will be modified in response to scores higher up the structure.

Phase **1303** for the generation of a list of associated preferred terms is detailed in *Figure 18*. At step **1801** a rule base is identified having a score greater than a predetermined threshold. Thus, for a particular application a threshold may be set at forty-eight percent. At step **1802** additional triggered

preferred data characteristics are identified by associating successful rule bases with parent categorisations by rule base links.

At step **1803** lists of successful and inferred rule bases are combined to form overall lists of preferred data characteristics. Step **1803** results in data being generated by a subsidiary processor, such as processor **311**, which is then supplied back to the central processing system **305**. Central processing system **305** is responsible for constructing a table of the type shown in *Figure 19* in which an entry is present for each preferred term. The specific preferred terms are stored in column **1901** and, for each of these terms, column **1902** defines a specific pointer to a position in memory associated with the central processing system **305**. Specific data files are identified by file names and the number of files associated with each preferred term is variable, depending on the nature and the amount of input data being considered. Thus, in order for this data to be accessible quickly while optimising use of the storage capacity within the central processing system **305**, an indication of the file names is stored in the form of a linked list as illustrated in *Figure 20*.

Referring to *Figure 19*, the preferred term "OIL_INDUSTRY" has been associated to a pointer 0F8912. Address 0F8912 is the first in column **2001** of the linked list. Column **2002** identifies a particular file name and column **2003** identifies the next pointer in the list. Thus, entry 0F8912 points to a particular file with the file name "OIL_INDUSTRY_NETHERLAND_3" with a further pointer to memory location 0F8A20. At memory location 0F8A20 a new file name is provided, illustrated at column **2002** and again a new pointer is present at column **2003**. Eventually, all relevant files will have been considered and the end of the list is identified by address 000000 at the pointer location in column **2003**.

In an active system, the database **323** will be continually updated and users will continually be given access to the database, all under the control of the central processing system **305**. Thus, with reference to *Figure 2*, it should

be understood that the association step illustrated at **203** and the searching step at **204** are actually concurrent and will be effected in response to the availability of data and the demand for searching respectively.

Procedures **204** for performing a search in response to a user request are detailed in *Figure 21*. At step **2101** a user logs onto the system and at step **2102** a search method is identified. At step **2103** search criteria are defined and at step **2104** the search criteria are processed to determine preferred terms. At step **2105** a list of preferred terms are supplied to the central processing system **305**.

At step **2106** a question is asked as to whether the host has responded and when answered in the affirmative titles of associated data files are displayed at step **2107**.

At step **2108** a question is asked as to whether the user wishes to view identified data and when answered in the affirmative the data is viewed; after being downloaded over the communication channel, at step **2109**.

At step **2110** a question is asked as to whether another search is to be performed and when answered in the affirmative control is returned to step **2102**.

Step **2102** requires the search method to be identified and in order to achieve this a user is prompted by a screen display of the type shown in *Figure 22*. Thus, a plurality of text boxes are presented to the user inviting the user to specify a search method.

Step **2103** for the defining of search criteria results in the user being prompted by a screen of the type shown in *Figure 23*. Terms providing a basis for the user's search are displayed in a window **2301**. Preferred terms are displayed in uppercase characters, such as the entry shown at position **2302**.

The displaying of titles of associated files at step **2107** results in the user seeing information displayed of the type illustrated in *Figure 24*. Each entry, such as entry **2401**, includes a check box **2402**. Check boxes **2402**

allow a particular item to be selected by a user such that the actual information file may be supplied to the user from the central database over a communication channel.

Claims

1. A method of associating files of machine-readable data with information types, comprising steps of

examining data elements in a file to identify occurrences of specified data types;

adjusting a score in response to said identified occurrences;

further adjusting said score in relation to a size of said data; and

associating said information types with processed files in dependence upon said score values.

2. A method according to claim 1, wherein said further adjustment is performed if the size of said data is below a predetermined threshold.

3. A method according to claim 1 or claim 2, wherein said further adjustment is performed up to a predetermined maximum weighting value.

4. A method according to any of claims 1 to 3, wherein said information types are expressed as preferred terms and are specified by information outlines.

5. A method according to claim 4, wherein said information outlines define preferred terms in a branching hierarchical structure and said adjustable values are defined at branches of said structure.

6. A method according to claim 5, wherein said further adjustment includes adjustment at a plurality of said branches.

7. A method according to claim 6, wherein said a score value is determined by multiplying values stored at branches throughout said

hierarchy.

8. A method according to claim 1, wherein said relation for adjusting said score is an inverse relation, such that said score is adjusted inversely in relation to a size of said data.

9. A method according to claim 8, wherein said size of said data is determined in terms of the volume of the stored file, representing the number of characters present in the file.

10. Apparatus for associating files of machine-readable data with information types, comprising

examining means for examining data elements in a file to identify occurrences of specified data types;

adjusting means for adjusting a score in response to said identified occurrences and for further adjusting said score in relation to the size of said data; and

associating means for associating said information types with processed files dependent upon said score values.

11. Apparatus according to claim 10, wherein said adjusting means further adjusts said score if the size of said data is below a predetermined threshold.

12. Apparatus according to claim 10 or claim 11, wherein said adjusting means is configured to perform said further adjustment up to a predetermined maximum weighting value.

13. Apparatus according to any of claims 10 to 12, wherein said associating means is configured to associate said information types and said

information types are expressed as preferred terms specified by information outlines.

14. Apparatus according to claim 13, including means for storing a branching hierarchical structure of preferred terms defined by said information outlines, wherein said adjusting means is configured to adjust values defined at branches of said structure.

15. Apparatus according to claim 14, wherein said adjusting means is configured to adjust values at a plurality of said branches.

16. Apparatus according to claim 15, including means for multiplying values stored at branches throughout said hierarchy to produce said score value.

17. Apparatus according to claim 10, wherein said adjusting means is configured to further adjust said score inversely in relation to a size of said data.

18. Apparatus according to claim 17, including means for determining the size of said data in terms of the volume of the stored file representing the number of characters present in the file.

19. A method of associating files of machine-readable data substantially as herein described with reference to the accompanying drawings.

20. Apparatus for associating files of machine-readable data substantially as herein described with reference to the accompanying drawings.

The Patent Office

25

Application No:	GB 9808807.3	**Examiner:**	K. Sylvan
Claims searched:	1-20	**Date of search:**	9 October 1998

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): G4A (AUDB)

Int Cl (Ed.6): G06F (17/30)

Other:

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
X	EP0822502 A1	BT. See page 7 lines 1-15.	1,10 at least
X	EP0364179 A2	NeXT. See the equation on page 4.	1,10 at least.
X	US5659766	Xerox. See column 6 lines 56-67, especially lines 62-63.	1,10 at least
X	US5598557	Caere Corp. See the equation in column 7.	1,10 at least

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

This Page Blank (uspto)